

N° 6

ISSN 0756-6246

# ORDI-5

TIREZ PLUS DE VOTRE SINCLAIR

**ESSAI :**



**LE MICRODRIVE**

**INITIATION :**

comment fonctionne  
un ordinateur

**LANGAGES sur ZX :**

Logo, Pascal, Forth, Assembleur

**12 JEUX - 15 PROGRAMMES**



M 2606-6-25 F

N° 6 - 1<sup>er</sup> trimestre 1984

Canada 5.95 \$ - Belgique 200 FB - Suisse 11 FS

**25 FF**



**DOSSIER  
HORS SÉRIE**

# L'ORDINATEUR INDIVIDUEL

*Ce dossier présente une sélection des meilleurs programmes publiés dans l'Ordinateur individuel*

## 50 PROGRAMMES BASIC

Jeux, enseignement, professionnels, utilitaires, etc. et un tableau comparatif de tous les Basic pour pouvoir utiliser les programmes sur : Alice, Apple, Atari 400, Atom, BBC, Commodore 64, DAI, Dragon 32, Hector, Lynx, MZ 80 A, Oric, PET/CBM, TO 7, TI 99, TRS 80, Vic 20, ZX 81, ZX Spectrum.

**en vente dans tous  
les kiosques  
à partir du 10 décembre**

### Editorial



La Tour de Babel est une expression qui commence à n'être plus vraie pour les ordinateurs familiaux du moins ceux que nous connaissons bien. Le ZX Spectrum et le ZX81 acceptent désormais grâce à des logiciels performants (il est vrai assez consommateurs de mémoire) de comprendre les programmes en Pascal, en Forth et même en Logo au-delà du BASIC. Sans parler bien sûr de l'assembleur qui n'est que la façon polie de s'adresser au microprocesseur que l'on peut interpeller directement par le langage machine.

Avec l'étude d'une cassette (Pascal), le démontage d'un jeu expliqué (assembleur et Forth), un simulateur de tortue (Logo), avec toutes ces choses ORDI-5 a choisi de vous initier aux secrets de ces différents langages par le chemin des écoliers (en récréation).

Encore de l'initiation : comment marche un ZX, quels sont les éléments essentiels d'un ordinateur, cela aussi figure en bonne place dans la revue avec bien sûr plein de jeux, plein de programmes, et un compte rendu complet sur le microdrive tant attendu.

Joyeux Noël et à bientôt.

PS : ORDI-5 aimerait bien voir ses petits souliers remplis de programmes 1K spirituels et bien tournés que vous aurez composés pour lui.

ORDI-5

\* Sinclair, ZX 80, ZX 81, ZX Spectrum sont des marques déposées.

### Sommaire

Courrier	10
Magazine	13
Librairie	15
Trucs à brac	17
Juvenilia	18
Le ZX fort en thème	19
Un ordinateur comment ça marche ?	22
Logigraph	24
Le microdrive	26
Modulons notre FORTH	28
PASCAL sur ZX	30
Port de sortie 8 bits	31
Côté court	34
Mon bel écran	34
Code machine	35
3 utilitaires ZX81	40
5 cassettes Spectrum	42
Le ZX fort en thème	43
Programmes de jeu	45

Rédacteur en chef : Alain Pinaud.

Editeur : Jean-Pierre Nizard.

Directeur de la publication : Bernard Savonet.

Conseiller technique :

Xavier Linant de Bellefonds.

Maquette : Sylvine Dautref.

Secrétariat : Nicole Alleman.

Illustration : Christian Augé, Yvan Boussion, Dominique Delpierre, Stéphane Druais, Isabelle Mounier. Couverture : Sylvie Taugourdeau sur une idée de Christian Augé.

Ont collaboré à ce numéro :

Kaarina Alain, Tristan D'Amico, Freddie Blin, Jean-Claude Bouman, Jean Casenave, Bruno Célerier, B. Clergeot, F. Delaquaize, F. Falque, Patrick Guculle, Didier Gumy, J.-C. Heur, Marcel Henrot, B. Lacoste, Morandini, Michel Murbach, Jean-Luc Papillon, G. Pedrolli, Pesce, Claude Remy, Sherer, J.-M. Schimpft, Régis Senegou.

Rédaction et abonnements :

Editrace, 8, rue Saint-Marc, 75002 Paris.

Régie publicitaire :

Force 7, Anne Jourdan,  
5, place du Colonel-Fabien,  
75491 Paris Cedex 10  
Tél. : (1) 240 22 01

Diffusion auprès des boutiques informatiques et des libraires :

PSI Diffusion,  
B.P. 86, 77402 Lagny/Seine-et-Marne Cedex.

# Abonnez-vous à ORDI-5

# VISMO

Vente Informations Services Micro-Ordinateurs  
22 Bd de Reuilly, 75012 PARIS  
Tél: (1) 628.28.00  
VENTE ET DEMONSTRATION DE 14 H à 21 H SAUF LUNDI  
Métros : Daumesnil ou Dugommier

**VISMO EXPRESS :** Livraison dans toute la France. Nous encaissons vos chèques à l'expédition de votre commande, jamais à la réception de vos ordres.

## SINCLAIR ZX SPECTRUM

Spectrum 16K Pal.....	1.480 F
Spectrum Péritel.....	1.850 F
Spectrum 48K Pal.....	1.965 F
Spectrum Péritel.....	2.325 F

### K7 JEUX (16 ou 48K)

Panique.....	75 F
Minedout.....	86 F
Space Invader.....	86 F
Androïde.....	75 F
3 D Tank.....	75 F
Météorites.....	75 F
Jawz.....	75 F
Fruit Machine.....	75 F
Gold Mine.....	75 F
Spawn Of evil.....	75 F
Road Toad.....	75 F

### K7 JEUX REFLEXION (16 et 48K)

Simulateur de vol.....	95 F
Othello (16 ou 48K).....	75 F
Awari (16 ou 48K).....	54 F
Echecs (48K).....	115 F

### K7 EDUCATION

Math (16 ou 48K).....	54 F
Histoire (16 ou 48K).....	54 F

### K7 GESTION

Directeur financier (48K) ..	125 F
Gestion de fichiers (16 ou 48K).....	115 F

### K7 UTILITAIRES

Pascal 4 T (48K).....	260 F
Devpac Assembleur/ Désassembleur (16K).....	160 F

### INTERFACES

Carte 8 E/5.....	395 F
Interface manette de jeux ..	250 F
Poignée de jeu.....	120 F
Modulateur UHF N/B.....	190 F
ZPS 84.....	895 F

## SINCLAIR ZX-81

ZX-81.....	580 F
Mémoire 16K.....	340 F
Imprimante.....	690 F

### K7 JEUX (16K)

Simulation de vol.....	95 F
Patrouille de l'espace.....	65 F
Phantom.....	75 F
Stock car.....	75 F
Invaders.....	65 F
Tyrannosaure Rex.....	75 F
Gulp.....	75 F
Biorythmes.....	85 F
Chiromancie.....	85 F
Scramble.....	75 F

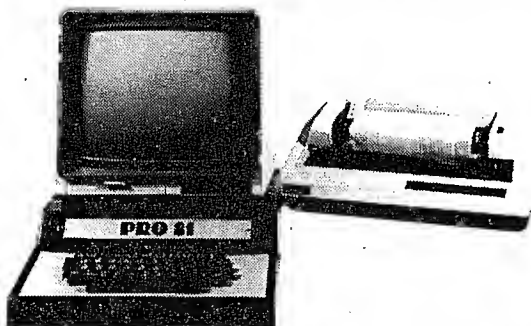
### K7 JEUX REFLEXION (16K)

Othello.....	95 F
Echecs.....	95 F
Tric-Trac (Backgammon).....	85 F
Awari.....	85 F

### K7 GESTION (16K)

Gestion compte bancaire ..	95 F
----------------------------	------

## SYSTÈME "CLÉS EN MAIN" DE COMPTABILITÉ GÉNÉRALE



# 6500 F TTC!

Voilà un prix vraiment attractif pour les artisans, commerçants et PME qui savent "faire leurs comptes"! Le système PRO 81 a été mis au point par des comptables et testé en entreprise. C'est aujourd'hui la façon la plus économique de traiter et de contrôler sa comptabilité.

Le système PRO 81 comprend :

- un ZX 81
- un boîtier
- un clavier mécanique
- un inverseur TV/Video
- un moniteur Zenith
- une extension mémoire 64 K
- un interface imprimante
- une imprimante 80 col.
- un programme de comptabilité générale sur K7.

Bien entendu les possesseurs de ZX 81 peuvent acheter séparément les compléments nécessaires. Nous consulter.

Vu - File.....	110 F	Pratique des Sinclair.....	80 F
Vu - Calc.....	110 F	Maîtrisez votre ZX-81.....	70 F
ZX-Multifichiers.....	150 F	50 Programmes pour ZX-81.....	32 F
Data-Base.....	60 F	Montages périphériques du ZX-81.....	32 F

### K7 UTILITAIRES (16K)

Assembleur Ardic.....	75 F	Pilotez votre ZX.....	63 F
Moniteur Désassembleur ..	75 F	ZX à la conquête des jeux ..	65 F
Tool Kit/Test.....	75 F	70 programmes ZX-81.....	60 F
Tool Kit II.....	90 F	ZX-Spectrum.....	60 F
ZX - Tri.....	75 F	Super carte couleur Pentron, connectable directement sur le ZX. Pas de soudure, nécessite une 16K Sinclair et une TV avec Péritel.....	450 F
Fast Load Monitor (16 ou 64 K).....	75 F	Magneto K7.....	430 F

### ZX-81

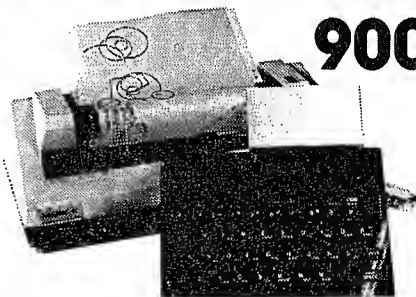
Comment programmer votre ZX; cours comprenant : 2 K7, un cours de programmation, un manuel d'exercices.....	195 F	Carte Auto-Repeat.....	95 F
Le petit livre du ZX.....	72 F	Clavier AB5.....	140 F
La pratique du ZX - T. 1.....	72 F	Carte Sonore.....	350 F
La pratique du ZX - T. 2.....	82 F	Interface Manette de Jeux ..	250 F
Etudes pour ZX-81 - T. 1.....	82 F	Manettes de Jeux.....	120 F
Etudes pour ZX-81 - T. 2.....	82 F	Carte 8 E/5.....	390 F
		Carte Mère.....	192 F
		Connecteur Femelle.....	40 F
		Alimentation I, 2A.....	180 F

POUR OBTENIR UNE QUALITÉ PROFESSIONNELLE AVEC VOTRE SPECTRUM:

## INTERFACE ZPS 84

Cet interface avec sortie parallèle Centronics permet de connecter à votre Spectrum une imprimante GP100 de Sekosha. De plus, il possède une sortie pour brancher un moniteur vidéo noir et blanc. (Bien utile si vous voulez faire de l'informatique à la campagne pendant le week-end sans avoir à emporter votre gros téléviseur!). L'interface ZPS 84 exécute directement les fonctions LPRINT, LIST qui travaillent automatiquement sur 80 colonnes. Également disponibles : la fonction COPY et la fonction TAB (tabulation). Les symboles graphiques propres au Spectrum sont reproduits correctement. L'interface permet un choix aisé entre

l'impression en mode standard, et l'impression en double largeur. La fonction COPY reproduit ce qui est sur l'écran, directement en haute définition graphique, dans une matrice de 256 x 176 points. Vous pouvez ainsi recopier intégralement les dessins que vous avez réalisés sur votre écran (et la reproduction est fidèle !... les cercles imprimés sont réellement ronds). Le logiciel du ZPS 84 est contenu dans une mémoire morte. Il est entièrement transparent pour l'utilisateur. Une fonction RENUMÉROTATION des lignes de programmes BASIC vous sera fort utile pour la mise au point de vos propres programmes. Très bientôt, il existera une carte TRACEUR DIGITAL qui pourra se loger dans cet interface. Accompagnée d'un bras articulé, elle vous permettra de copier sur votre écran une image dont vous suivrez les contours. Une fois la copie faite, vous pourrez la reporter facilement sur l'imprimante par l'ordre COPY. Une vraie petite merveille, qui l'Documentation gratuite sur demande.



# 900 F (TTC)

GP 100 A.....	2.300 F
Listing Blanc (bande carrol).....	230 F
Câble Imprimante GP 100 A.....	170 F
Moniteur Zenith Monochrome.....	1.050 F
Imprimante GP 100 A.....	2.300 F
Imprimante GP 50 A.....	1.450 F

### PACK VISMO

GP 100 A + Interface Memotech - Câble + 1000 Feuilles Listing.....	2.900 F
GP 100 A + ZP-82 + 1000 Feuilles Listing.....	3.100 F

### EXTENSIONS MEMOTECH

Mémoire 16 K.....	360 F
Mémoire 32K.....	540 F
Mémoire 64K.....	790 F
Haute Résolution Graphique.....	490 F
Memotext.....	440 F
Z 80 Assembleur.....	440 F
I/F Centronics.....	440 F
Clavier Mécanique.....	540 F
RS 232.....	640 F
Mémocalc Analyse.....	440 F

### PERIPHERIQUES

Nouveau Moniteur Zenith écran vert (très beau).....	1.050 F
---	---------

**Demandez les produits Vismo chez votre revendeur habituel.**

**POUR DÉTAXE A L'EXPORTATION**  
**SERVICE COMMANDE EXPRESS CRÉDIT RÉCLAMATIONS**  
**TÉLÉPHONE 586.60.10**

## BON DE COMMANDE

à retourner à Vismo, 68 rue Albert 75013 Paris

Nom : \_\_\_\_\_ Prénom : \_\_\_\_\_

Adresse : \_\_\_\_\_

\_\_\_\_\_ Ville : \_\_\_\_\_

Code Postal : \_\_\_\_\_ Tél. : \_\_\_\_\_

Date : \_\_\_\_\_ Signature : \_\_\_\_\_

☐ Désire recevoir catalogue 20 F (remboursable à la 1<sup>re</sup> commande)

LED  
M7  
MS  
OI

Quantité	Désignation	Prix unit. TTC	Prix total TTC

MODE DE REGLEMENT  
☐ Chèque bancaire joint  
☐ CCP joint  
☐ Mandat-Hellre joint  
☐ Contre-remboursement

☐ Participation frais de port et d'emballage + 30 F  
☐ Port gratuit pour + de 3000 F d'achat sous 90 jours.  
☐ Contre-remboursement + 30 F.



# ORDI-5

## LE MAGAZINE DES UTILISATEURS DE SINCLAIR



Si vous utilisez un ordinateur SINCLAIR (ZX 81, ZX 80 ou Spectrum) ou si vous comptez en acheter un, sachez que la revue **ORDI-5** a été créée pour vous. Indépendant de tout constructeur ou importateur, **ORDI-5** vous fournit quatre fois par an des programmes, des conseils, des astuces, de nouvelles idées d'utilisation. **ORDI-5** teste pour vous en toute objectivité et indépendance les

produits matériels et logiciels adaptables sur votre SINCLAIR. **ORDI-5** vous tient au courant de toutes les nouveautés susceptibles de vous intéresser.

**ORDI-5** n'est pas en vente chez les marchands de journaux. Pour vous abonner ou pour recevoir 1 numéro, il vous suffit de nous retourner le bon de commande ci-dessous.

**Abonnez-vous, vous économiserez 20%**

## ORDI-5, le complément indispensable de votre ZX

\*marques déposées



### BON DE COMMANDE

à retourner à ORDI-5, 8 rue Saint-Marc 75002 PARIS

Nom \_\_\_\_\_ Profession \_\_\_\_\_

Adresse \_\_\_\_\_

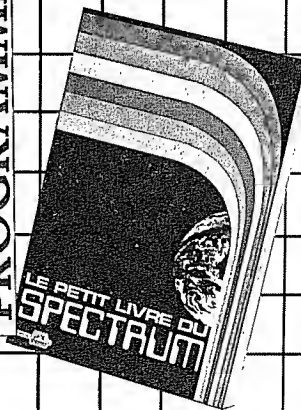
Pays \_\_\_\_\_ Code postal \_\_\_\_\_ Ville \_\_\_\_\_

☐ Je désire recevoir le n° 1 ☐ le n° 2 ☐ le n° 3 ☐ le n° 4 ☐ le n° 5 ☐ le n° 6 ☐ de ORDI-5 (prix d'un n° 25 FF ; Etranger\*\* 30 FF ; par avion 40 FF).

☐ Je désire m'abonner à ORDI-5 pour 1 an, 4 n°s, à partir du n° 1 ☐ du n° 2 ☐ du n° 3 ☐ du n° 4 ☐ du n° 5 ☐ du n° 6 ☐ (tarif France 80 FF ; Etranger\*\* 90 FF ; par avion 160 FF). (Actuellement ORDI-5 est trimestriel).

Ci-joint mon règlement indispensable par chèque bancaire ☐ chèque postal ☐ virement ☐

\*\* Pour les pays autres que la France, utiliser un virement en FF compte Crédit Lyonnais Paris n° 30002 00402 8455 J. Les frais de virement sont à la charge de l'acheteur



**ACHP**





Depuis quelque temps, il semble bien que la curiosité de nos lecteurs évolue nettement vers la constitution interne du ZX81, ainsi que vers le raccordement d'accessoires. Indépendamment des articles paraissant dans nos colonnes sur ces sujets passionnants (surtout compte tenu de l'auréole de mystère dont les entoure Sinclair), le meilleur ouvrage que nous ayons eu l'occasion de lire sur la question est sans conteste : "THE INS AND OUTS OF THE TIME TS 1000 AND ZX81", écrit par Don Thomasson et publié par Melbourne House. Il est hélas fort difficile de se procurer en France ce petit livre dont nous avons fait l'analyse dans ORDI-5 n° 4.

Voici les coordonnées de son éditeur, qui ne refuse guère les commandes émanant de l'étranger :

MELBOURNE HOUSE Publishers Ltd.  
Glebe Cottage Glebe House  
Station Road Cheddington  
LEIGHTON BUSSARD  
BEDFORDSHIRE LU7 7NA (Grande Bretagne)

Un mandat-poste international est la solution la plus commode pour régler le montant qui sera réclamé avant expédition de l'ouvrage.

## UN CIRCUIT INTEGRE PROPRE A SINCLAIR

Notre lecteur Pascal Aillerie trouverait par exemple dans ce livre l'explication de son échec lorsqu'il a tenté de faire accéder la machine à un nouveau générateur de caractères implanté par lui en RAM. Nous avons vu, en effet dans notre article du n° 5, que la modification du contenu de registre I n'autorise un changement de jeu de caractères que sur l'imprimante, à condition encore d'ajouter un petit montage additionnel.

La raison de tout cela tient à ce que le générateur de caractères de la ROM est adressé, lors de la création d'une image TV, non par le microprocesseur ZX80, mais par le fameux "circuit intégré Sinclair", tandis que le ZX80 lui, travaille en liaison avec la RAM ! C'est d'ailleurs pour cela que neuf résistances séparent la ROM de la RAM sur les lignes A0 à A8 du bus d'adresses. Monsieur Vincent Lacour trouvera encore bien d'autres informations dans ce livre, sur les autres fonctions du circuit Sinclair, qui l'intrigue plus particulièrement. Ce gros circuit intégré appartient à la famille des "réseaux prédiffusés" ou, en anglais, "uncommitted logic arrays", souvent abrégé en ULA.

Il s'agit de réseaux de portes logiques dont on peut modifier l'interconnexion au moyen de procédés industriels simples, tout comme on programme une ROM. Il est donc tout à fait vain de chercher les caractéristiques de ce circuit dans un quelconque recueil de produits standards, puisqu'il n'est destiné qu'au seul ZX81.

Seules des recherches patientes, confirmées par des recoupements avec le schéma du ZX80 (ancêtre du ZX81

réalisé à partir de boîtiers TTL courants), peuvent conduire à une compréhension satisfaisante du fonctionnement de ce circuit chargé de bien des tâches :

- génération de la fréquence d'horloge générale, au moyen d'un filtre céramique de récepteur TV (6,5 MHz).
- division par 2 pour obtenir l'horloge à 3,25 MHz du microprocesseur Z80 A.
- gestion des entrées-sorties cassette
- gestion du clavier
- sélections mémoire
- génération des lignes de l'image TV, en liaison avec le générateur de caractères de la ROM.

## A PROPOS DE L'IMPRIMANTE

Beaucoup de questions également sur l'imprimante Sinclair : comme tous les produits de la marque, micro-drive compris sans oublier le téléviseur ultra-plat, ce périphérique fait appel à des techniques très anticonformistes.

Une sorte de courroie intérieure supporte deux petites pointes élastiques défilant à grande vitesse, mais en parfait synchronisme avec l'avance du papier, puisque les deux mouvements sont dus à un même moteur lié à un train d'engrenages nylon.

Comme ces pointes sont diamétralement opposées, l'une attaque un bord du papier juste au moment où la précédente quitte l'autre.

On obtient ainsi un balayage permanent du papier métallisé, fort similaire à celui d'un image TV.

C'est d'ailleurs pour cela que cette imprimante arrive à recopier fidèlement l'écran TV, même en présence de graphismes haute ou basse résolution, ce qui est sans précédent compte tenu du prix pratiqué !

Comme la mécanique est bien loin de suivre l'électronique côté vitesse, imprimante et ordinateur échangent en permanence des signaux de synchronisation par le port 251 ou FB. Un avantage peu connu de cette petite imprimante est son incomparable fiabilité en matière de transfert d'information : comme les données d'impression lui parviennent point par point, chaque caractère imprimé nécessite huit octets, répartis sur huit lignes successives.

Si l'un de ces octets se trouve perdu ou altéré, il n'en résultera qu'une légère déformation du caractère, alors que sur une imprimante recevant directement les codes ASCII (un seul octet par caractère), le caractère serait irrémédiablement perdu.

Le principal inconvénient de ce petit périphérique est la nature du papier employé, absolument inutilisable pour la correspondance ou de la facturation.

Il existe, du moins en Grande Bretagne, une foule d'interfaces permettant de raccorder des imprimantes papier ordinaire au ZX81.

Le prix de ces accessoires varie beaucoup selon que le logiciel d'accompagnement est logé dans une ROM incorporée, ou tout simplement fourni sur cassette. Tout un transcodage est en effet indispensable entre les codes Sinclair, et les octets ASCII que réclame, par exemple, une imprimante conforme au standard CENTRONICS.



En France, nous ne disposons guère que de l'interfaçage MEMOTECH, et d'un adaptateur spécialement étudié par TEKELEC pour son imprimante "best seller", la GP 100 SEIKOSHA.

Il s'agit du nec plus ultra, permettant au ZX81 de traiter les minuscules et même certains caractères graphiques, mais cette interface coûte... deux fois le prix du ZX81 et l'imprimante près de cinq fois ! Autant dire qu'il est plus sage de changer d'ordinateur, ce qui se ressentira par la même occasion au niveau du clavier, bien peu adapté à des applications "bureautiques".

LA PERITELEVISION

Plusieurs questions de vidéo également, émanant de lecteurs souhaitant connecter leur ZX81 à un magnétophone, ou un téléviseur muni d'une prise "péritélévision" non standard à un providentiel ZX Spectrum. Dans le premier cas, rien n'est plus simple puisque le ZX81 peut être considéré exactement comme une antenne TV délivrant une émission sur le canal 36 (environ). Pour une adaptation directement en vidéo, on se reportera plutôt à cette rubrique parue dans notre numéro 2 : ce genre de problème avait, à l'époque, été traité dans le détail.

Pour ce qui est des prises "péritels" non conformes à la norme "SCART" en vigueur actuellement, signalons que des fabricants réputés de connecteurs proposent aux revendeurs TV toute une gamme de cordons pré-câblés capables de résoudre la plupart des problèmes, et même d'adapter, sans la moindre intervention interne, une prise normalisée à tout récepteur n'en comportant pas (coût 500 F environ)..

Tout bricolage dans ce domaine demande une grande prudence, car les variantes sont innombrables pour les prises non normalisées.

L'assistance d'un agent de la marque du récepteur est donc vivement souhaitable.

A titre de rappel, nous reproduisons cependant ci-après le brochage des prises péritélévision normalisées :

Patrick Gueulle

→ Dans le dernier numéro d'ORDI-5 vous avez publié un dessin intitulé MARTHE. Pourriez-vous m'envoyer le listing de ce programme ? (Joël Ganzin).

\* Derrière l'image de Marthe, il n'y a pas un programme particulier mais une table de points PLOTtée. Pour PLOTter Marthe, sans jeu de mot vulgaire, son créateur l'a d'abord dessinée sur du papier millimétré et a ensuite introduit une table dans l'ordinateur, chacun des points ayant une coordonnée horizontale et une coordonnée verticale - il existe des calques millimétrés qui vous permettront de transcrire ainsi n'importe quel motif graphique qui vous paraîtra beau.

→ Quel dommage que le Spectrum n'ait pas une télécommande de magnétophone ! il doit bien y avoir un moyen de décoder les ordres de chargement et de fin de programme (Bernard Breton).

\* Si le Spectrum n'est pas fourni avec un jack de télécommande de magnétophone c'est que, dès sa conception on a imaginé que la meilleure mémoire de masse pour lui ne serait pas une cassette mais une micro-disquette, laquelle s'arrête d'elle-même dès la fin de tout chargement.

→ Comment distinguer tout de suite si vos programmes sont faits pour le ZX81 ou pour le Spectrum ? Vous ne dites rien en tête d'article (Yves Dusonchet).

\* La différence est immédiate : lorsqu'un programme comporte plusieurs instructions par ligne ou des minuscules, c'est un programme Spectrum.

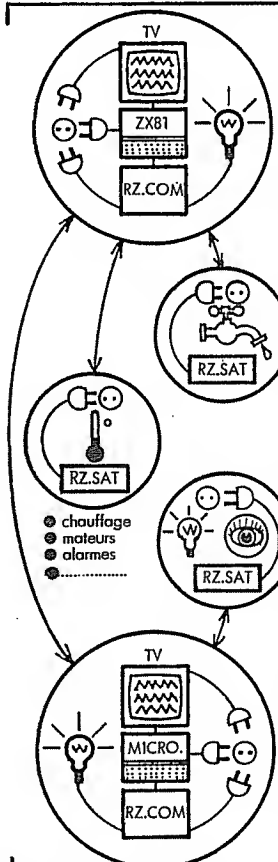
→ Je suis au regret de ne pouvoir renouveler mon abonnement à ORDI-5, ayant renoncé à acquérir un Spectrum devant le peu d'enthousiasme déployé par son importateur (G. Lepesant).

**nouveau** **ZX81 et tous micro ordinateurs.**

*votre ordinateur devient domestique!*

**Télécommande sans câblage...**

RZ.COM réseau informatique de communication de télémesure et de télécommande par courant porteur



RZ.COM et ses satellites RZ.SAT associés à distance, permettent de commander des appareils électriques (lampes, radiateurs, moteurs, électrovannes, sirènes, pastes radia, etc), effectuer des mesures de paramètres variés (lumière, humidité du sol, température, potentiomètre, etc) et communiquer avec d'autres ordinateurs (ZX81 au liaison RS232 à 300 bauds).

Un ensemble de plusieurs ZX81 et RZ.COM, et leurs satellites RZ.SAT permettent de constituer un véritable réseau informatique réalisant des automatismes variés programmables en BASIC, sans aucun câblage, par simple branchement sur des prises de courant ordinaires jusqu'à une distance de 150 m.

RZ.COM se présente dans un boîtier moulé (155 x 90 x 45 mm), relié au connecteur arrière du ZX81 et possède sa propre alimentation. Le ZX81, programmé en BASIC, lui transmet des commandes et en reçoit les réponses sans la forme de chaînes de caractères.

RZ.COM est constitué de :

- 1 calendrier perpétuel programmable : an, mois, jour, heure, minute et seconde et correction des dérives,
- 1 prise 220 V permettant de commander tout appareil électrique jusqu'à 1 KW.
- 1 commutateur à deux positions faisant office d'entrée logique programmable.

RZ.SAT possède le même équipement plus :  
 — 1 indicateur (LED) programmable,  
 — 1 entrée analogique liée à une cellule photo-électrique (ou d'autres capteurs : température, humidité du sol, potentiomètre, livrés dans une pochette séparée).

Notice et exemples : enveloppe timbrée et adresse

**BON DE COMMANDE** à retourner à :  
 — MINISYSTEMES — B.P. 30 — 13090 LUYNES

Je désire recevoir, avec manuel et exemples, par poquet poste recommandé :

— RZ.COM (ZX81) .....	<input type="checkbox"/>	980 FF : .....
— RZ.COM (RS232) .....	<input type="checkbox"/>	980 FF : .....
— RZ.SAT .....	<input type="checkbox"/>	790 FF : .....
— Pochette capteurs .....	<input type="checkbox"/>	120 FF : .....

(gratuite dans 1 kit RZ.COM + RZ.SAT)  
 — Frais d'expédition : 29 FF

Je paie par C.C.P. ou chèque bancaire de ..... libellé au nom de MINISYSTEMES, et joint au présent bon de commande.

Si je ne suis pas entièrement satisfait, je suis libre de retourner le matériel sous quinze jours, je serai alors totalement remboursé.

NOM :  
 ADRESSE :

SIGNATURE :

(ou pour les moins de 18 ans, de l'un des parents)

\* Cher Monsieur, nous citons votre lettre car elle est caractéristique d'un très abondant courrier : l'extrême lenteur du Spectrum à pénétrer en France est plus le fait de la fabrication que de l'importateur. Le grand succès de cet ordinateur dans les pays Anglo-Saxons a conduit Sinclair à considérer le marché français comme résiduel dans les premiers temps. Cet état de choses devrait changer dans le tournant 83-84 pour notre soulagement à tous.

→ J'aimerais savoir si l'on peut vendre ou acheter des cassettes du commerce à d'autres personnes - les vraies cassettes, bien sûr ? (Gérard Demoustier).

\* Certainement, seule la duplication de cassette est interdite car elle signifie reproduction et que la propriété littéraire et artistique qui protège les programmes s'oppose à la reproduction. Quand vous achetez une cassette de programme ce n'est pas le programme que vous achetez mais le droit à l'usage de ce programme : vous pouvez parfaitement revendre ce droit à usage à quelqu'un d'autre. Mais la plupart du temps n'est-ce pas, avant de céder la cassette originale n'est-on pas tenté de la dupliquer pour ses propres besoins ?...

Si vous voulez en savoir plus sur la protection juridique des programmes, vous pouvez consulter "L'Informatique et le Droit" de Xavier Linant de Bellefonds Que sais-je, n° 1923 (c'est un ami !).

→ Pourquoi ne parlez-vous pas davantage du Cyborg que la publicité présente comme la disquette adaptée au Sinclair ?

\* Nous attendons pour vous parler du système Cyborg que le système soit effectivement disponible, ce qui à notre connaissance n'est toujours pas le cas malgré les promesses faites à ceux qui ont payé d'avance. Nous ne pouvons en effet vous parler que des matériels que nous avons jugés sur pièces, ce qui nous l'espérons sera sans doute le cas bientôt.

## REGION ALSACE

**POUR Zx80/Zx81**

**CARTE MULTIFONCTION C.I.T.**

avec 8K de ROM supplémentaire

- SON sur T.V. (3 octaves)  
au clavier, par programme ou préenregistrée
  - HAUTE RÉOLUTION GRAPHIQUE  
sur impr. Sinclair axes gradués et quadrillage
  - 10 PAGES ÉCRAN EN MÉMOIRE  
écriture/lecture priorité au programme  
Inversion vidéo par soft partielle/totale
  - INTERFACE IMPRIMANTE TYPE CENTRONICS...  
Renumerotation du basic au pas choisi
  - PROGRAMMATION D'EPROM  
Enregistrement de programmes en basic,  
+ data, en assembleur.  
Affichage des registres Z80 en hexa,  
binaire, variables Sinclair en décimal  
Ecriture géante, scroll inverse, clear partiel,  
sirène, adresse en zone variables, etc. etc. ...  
Pas de réservation de mémoires, pas de chargement !  
Messages d'erreur sonores et littéraux
- Prix avec manuel d'utilisation T.T.C. 685.- FF**  
C.I.T. 4, rue de Bâle  
68300 SAINT-LOUIS Tél. (89) 67.76.01

**c.i.t.**

## PROTO-PACK ZX 81

**FINI L'ANGOISSE DES TRANSFERTS K7,  
LA FIABILITE DU PROFESSIONNEL A 100 % !**

**Ce module de mémoire permanente modifiable à volonté n'utilise aucune pile et peut contenir 5 longues minutes de k7 en un ou plusieurs programmes indépendants les uns des autres dans un même module, un simple ordre basic suffit à appeler un programme en une fraction de seconde sans qu'il se perde en route, à la vitesse de 160 k par seconde!!!**

**MADE IN FRANCE : PROTO EXPRESS tel de 8 à 20h : (6) 437.80.70**

**BON D'ESSAI GRATUIT**

A ADRESSER A

**PROTO EXPRESS BP 104 77003 melun cedex**

**je souhaite essayer cette nouvelle mémoire 15 jours sans engagement, si elle ne me convient pas il me suffit de vous la retourner pour que me soit restituée la caution.**

**PROTO-PACK pour RAM de .....K**

- ☐ avec liaison directe sur le ZX : **390,00 FF**
- ☐ avec liaison pour carte-mère : **320,00 FF**
- ☐ port contre remboursement : **+ 30 FF**
- ☐ paiement (ou caution) ci-joint, port gratuit.

NOM, ADRESSE ET SIGNATURE :

**8 LIVRES  
POUR VOTRE  
ZX-81**



par Trevor Toms

**136 pages - 72,00 FF/555,00 F8**

## par Jean-François Sehan

**96 pages - 82,00 FF/635,00 F8**

## par Xavier Linant de Bellefonds

**128 pages - 72.00FF/555.00FB**

par Marcel Henrot

152 pages - 82.00 FF/635.00 F8

par Jean-François Sehan

160 pages - 82.00 FF/635.00 F8

## par Jean-François Sehan

176 pages - 82,00 FF/635,00 F8

**NOUVEAU**

**128 pages - 35,00 F/250,00 F8**

Sinclair ZX 81 et ZX Spectrum -  
Timex 1000, 1500 et 2000

**240 pages - 102,00 FF/785,00 F8**

**Prix valables jusqu'au 31.12.83**



**au Canada**  
SCE Inc.  
65, avenue Hillside  
Montréal (Westmount)  
Québec H3Z1W1  
Tél. : (514) 935-1314

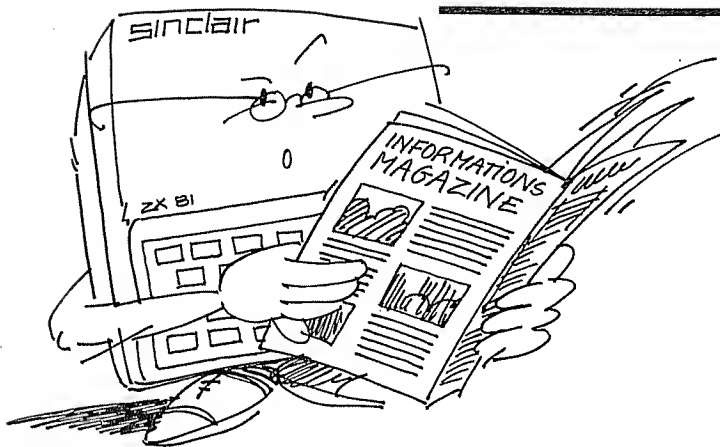
Code postal \_\_\_\_\_ Ville \_\_\_\_\_

Signature (obligatoire pour paiement  
par carte de crédit)

Code postal \_\_\_\_\_ Ville \_\_\_\_\_

Ville \_\_\_\_\_





# Magazine

## QUID NOVI SPECTRUM ?

Un 16 bits Sinclair ? On sait de bonne source que la firme est en train de préparer pour l'horizon 86 un ordinateur 16 bits. Le ZX80, le ZX81, le ZX Spectrum étaient tous basés sur un micro processeur à 8 bits, c'est-à-dire capable d'adresser seulement 64K de mémoire. Avec 16 bits cette limitation n'existe plus et le gain de rapidité est sensible. Déjà on peut considérer que les ordinateurs de table qui n'ont que 8 bits sont quelque peu périmés.

Un ZX83 ? Il en est question mais les informations sont assez contradictoires à ce sujet. Il pourrait s'agir d'un micro ordinateur portable conçu autour du microprocesseur 8051 d'Intel et d'un écran plat.

La batterie des extensions maison du Spectrum s'est récemment enrichie avec une deuxième interface, l'interface dite "2" qui permet de transformer l'ordinateur en système de jeu particulièrement souple par l'enfichage de cartouches ROM préenregistrées. Cette formule, qui a été imposée par le marché américain, procède d'une toute autre philosophie que l'interface "1" que nous analysons dans ce numéro. Conformément à la "loi Sinclair" les cartouches préenregistrées sont les plus petites du genre : elles ressemblent pour la taille à des extensions MEM telles qu'on les enfiche sous les ordinateurs de poche japonais. Le seul problème est que chaque cartouche, ayant fait l'objet d'une étude poussée, coûte évidemment cher à l'achat, environ 200 F, ce qui est un prix ordinaire comparé à d'autres constructeurs mais un prix élevé pour les produits Sinclair. L'interface n° 2 supporte évidemment les manettes de jeu.

Vous aimeriez faire lire directement par votre Spectrum les programmes que vous avez écrits pour un ZX81 sans avoir à tout retranscrire à partir du clavier avec les modifications nécessaires ? Une cassette de Amersham Software limited vous y autorise : il suffit de charger le Spectrum avec cette cassette pour qu'il lise automatiquement les signaux en provenance d'une cassette ZX81. Cette cassette n'est pas coûteuse : 4 livres seulement en Angleterre.

Memotech qui a fait des extensions nombreuses pour toute la série ZX81 a décidé de se lancer dans la fabrication des ordinateurs et propose un MTX 500 ; un appareil certainement très complet mais combien plus cher que le Spectrum !

MASH va commercialiser pour la rentrée deux nouveaux produits pour ZX81 : une mémoire RAM 16K (Econotech) 345 F TTC (cette mémoire a été étudiée pour éviter toute rupture de contact) et une mémoire RAM 64K ultra plate aux connecteurs professionnels plaqué or. Ces produits sont diffusés par la société INNELEC.

INNELEC distribue également en France les produits Basicare que nous avons déjà décrits dans ces colonnes et qui se présente, d'une manière assez professionnelle sous forme d'empilements (gamme organic micro) une dizaine et plus d'extensions peuvent ainsi se superposer avec des mémoires permanentes, des EPROM, des unités entrées sorties, etc.

Un nouveau distributeur pour les produits Sinclair : Silicone Valley à Nantes et à Angers.

## CA VA BIEN MERCI !

Le 13 octobre dernier le microdrive était présenté à la presse informatique française par Nigel Searle, le Directeur général de Sinclair Research. A cette occasion Direco International, le représentant exclusif en France de la firme britannique, en la personne de son président Eric Bompard, a informé les journalistes présents que 132000 micro ordinateurs familiaux ZX avaient été vendus. Ceci correspond d'après lui à plus de la moitié du marché français dans l'informatique familiale. Une autre information intéressante : une opération pourrait être montée visant à doter une centaine de lycées et collèges de machines ZX.

La société Sinclair pèse à ce jour 600 millions de francs de chiffre d'affaire ce qui est 2 fois plus qu'en 1982.

## EXPRESSION ECRITE

L'imprimante Sinclair est encore, 2 ans après sa conception parfaitement dans la course pour le rapport qualité/prix ; elle est si bon marché que de nombreux autres ordinateurs aimeraient bien pouvoir se l'adjindre. C'est possible maintenant, du moins pour les ordinateurs construits autour du microprocesseurs 6502 grâce à une interface proposée par Microtonic Computers systèmes LTD pour environ 350 F.

Maintenant que l'interface RS 232 est disponible pour le Spectrum il va être possible de regarder du côté des imprimantes un peu plus sophistiquées que celle du constructeur sans rechercher pour autant des interfaces encombrantes en provenance de fournisseurs plus ou moins réputés. Il existe chez Brothers une jolie petite imprimante ultra plate l'EP 22 avec interface incorporée qui coûte en France moins de 3000 F. Nous essaierons de savoir si elle est compatible avec le Spectrum.

Un traitement de texte professionnel pour Spectrum 48 K avec justification à droite et 64 caractères par ligne ? C'est ce que propose le groupe Sémaphore, consultants en communications. Naturellement ce logiciel est compatible avec la majorité des imprimantes professionnelles.

Pour ceux qui voudraient utiliser leur Spectrum directement interfacé à une imprimante de qualité sans attendre la commercialisation de l'interface 1 sur le continent, il est possible de se procurer chez KEMPSTON Microelectronics pour le prix de 500 F une interface RS 232 munie de son câble : cette interface à la taille d'une extension mémoire 16K de ZX81.

COMBIEN LA PUCE ?

Bientôt l'octet de mémoire vive à un centime ? La chose est désormais en vue avec les nouvelles additions mémoire pour ZX81 qui sont commercialisées en Angleterre pour moins de 16 livres, par Microcare.

La différence financière entre un Spectrum 16K et un Spectrum 48K est en passe de devenir négligeable : Fox electronics propose un kit avec une dizaine de puces à fixer soi-même à l'intérieur du Spectrum pour 20 livres anglaises.

AUX QUATRE Q.I. DU MONDE

Un ZX parlant arabe ? Cela existe, c'est un négociant de Jeddah qui le commercialise en Arabie Séoudite : les modifications de la MEM ont dû être radicales ; d'une part en Arabe les lettres n'ont pas le même graphisme selon qu'elles se trouvent au début, au milieu ou la fin des mots et ensuite et surtout, ce que personne n'ignore, l'Arabe s'écrit de droite à gauche, l'ensemble des fonctions d'éditions est donc revu.

Sinclair va fournir des ZX81 et des Spectrums en Kits à la South China Computer Company (GUANGZHOU-Canton). Les premiers engagements portent sur un marché de près de 10 millions de livres.

## CLUBS

- Microtel ADEMIR TREMBLAY, Collège Romain Rolland, Rue de Reims, 93410 VAUJOURS (Sec. Pascal VIEVILLE).
- ZX Microclub Résidence Europa, 6 Bd Léopold III, B-1030 BRUXELLES (Sec. Michel Hunin).
- Club Informatique de Conflans, Collège Jules Ferry, CONFLANS (Sec. Mme Mlet).

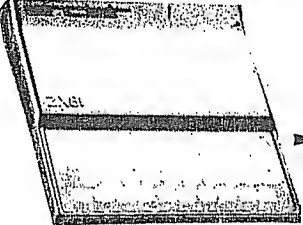
## ADRESSES DES SOCIETES MENTIONNEES

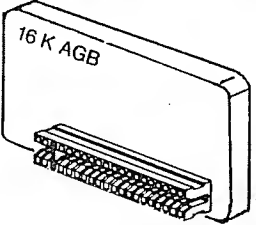
- ARAB RAM, RAMEZ HALABY & CO, P.O. BOX 147, JEDDAH, Arabie Séoudite.
- MICROCARE, 79 Thane Works, Thane Villas, LONDON N7, GB.
- FOX ELECTRONICS, 141 Abbey Road, Basingstoke, HANTS, GB.
- AMERSHAM SOFTWARE LTD, Long Roof, Hervines Rd, Amersham, BUCKS, GB.
- MEMOTECH, Memotech Limited, Station Lane Industrial Estate, Witney, Oxon, O78 6BX, GB.
- SEMAPHORE, JP Cardinaux, CH 1283 LA PLAINE (GENEVE) Suisse.
- E.G. de la M.INFO, EGNI, 20 Bd St Denis, 75010 PARIS.
- INNELEC, 110 bis Av. du Général Leclerc, 93500 PANTIN.
- SILICONE VALLEY, 87 quai de la Fosse, NANTES 7 rue Boisnet, ANGERS
- MICROTANIC COMPUTERS SYSTEMS, 16 Usland Rd, Dulwich, LONDON SE 22, GB.
- KEMPSTON Microelectronics, Unit 30, Singer Way, Woburn Road, I.E., KEMPSTON, Bedford NK 42 7AF, GB.

# SINCLAIR ZX'81 AGB - IS'

LA 1<sup>re</sup> GAMME DE MATERIELS ET LOGICIELS POUR VOTRE ZX 81  
EN DIRECT DU CO-STRUCTEUR, A VOS MEILLEURS PRIX

Si vous avez des questions n'hésitez pas à nous contacter au (38) 72.25.95. Nous serons heureux de pouvoir vous répondre.

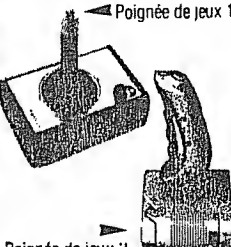




**PROMOTION**

Offre valable jusqu'au 31/1/ 84

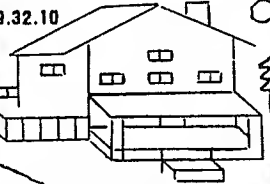
**= 850 F\* au lieu de 940 F**



INTERFACE parallèle ZX 81	249
INTERFACE parallèle SPECTRUM	299
INTERFACE série ZX 81	289
INTERFACE série SPECTRUM	319
CABLE INTERFACE (à préciser)	150
CARTE 2 supports EPROM et RAM 6116 ZX 81	199
TOUCHE Repeat ZX 81 KIT	50
CLAVIER ABS	140
CARTE GRAPHIQUE montée, compatible, toutes mémoires, se programme en BASIC	179

CARTE SONORE montée avec ampli compatible, toutes mémoires, se programme en BASIC

écoutez-la au (38) 39.32.10



Dessins obtenus avec notre carte graphique

POIGNEE DE JEUX 1 : la paire ..... 150

Stock limité



POIGNEE DE JEUX 2 : pièce .. 120

le nec plus ultra (4 ventouses, possibilité de jouer avec une seule main)

CARTE POIGNEES DE JEUX ne nécessite

aucune modification programme.

ZX 81 ..... 179  
SPECTRUM ..... 199

Documentation gratuite contre 2 timbres à 2 F

**ATTENTION  
NOUVELLE  
ADRESSE**

**BON DE COMMANDE Tél. (38) 72.25.95**

à retourner à **A.G.B.** - Les 4 Arpents -

23, rue de la Mouchetière, Z.I. d'Ingré, 45140 St-Jean-de-la-Ruelle

Nom ..... Prénom .....

Adresse .....

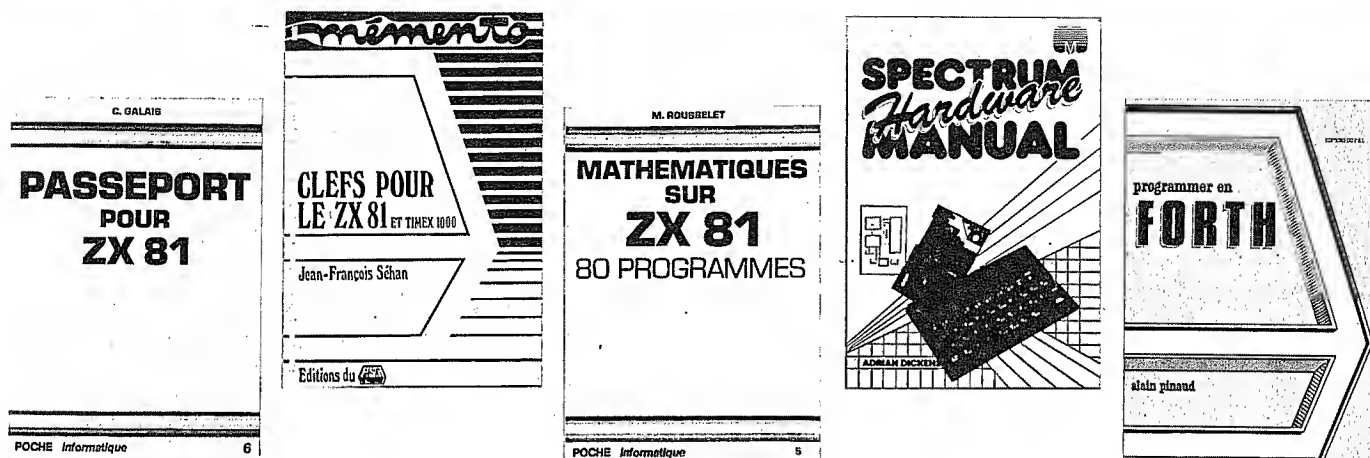
Ville .....

Code postal ..... Tél. ....

Date ..... Signature .....

Quantité	Désignation	Prix unit. TTC	Prix total TTC
MODE DE REGLEMENT		Participation frais de port et d'emballage : 20 F	
Cheque bancaire joint		80 F pour le ZX 81	
CCP joint		Contre-remboursement : x 1	
Mandat-lettre joint			
Contre-remboursement			

1. Marques déposées



## PASSEPORT POUR ZX81

Par Claudy Galais  
Collection Poche Informatique  
Prix : 50 F environ  
128 pages

Sixième titre de la toute jeune collection "POCHÉ Informatique ETSF", PASSEPORT POUR ZX81 doit avant tout être considéré comme un ouvrage de référence. Il s'adresse à notre avis surtout aux habitués d'autres machines souhaitant se familiariser rapidement avec les particularités du BASIC SINCLAIR.

Vu sous cet angle, le classement alphabétique des mots-clé se justifie par un souci de commodité d'emploi, alors qu'il aurait constitué une faute de conception dans un ouvrage de pure initiation, dont l'approche doit être très progressive. Chaque mot-clé est illustré par un petit programme d'une originalité certaine : généralement dénué de toute utilité pratique, il est conçu de manière à utiliser le mot-clé étudié, dans tous les cas susceptibles de se présenter. Toutes les variantes de mise en oeuvre étant ainsi passées en revue, les surprises ultérieures se trouvent éliminées de façon élégante.

Le débutant ayant appris le BASIC au moyen du ZX81 trouvera dans PASSEPORT, grâce à sa présentation "index", les réponses à ses petits problèmes bien plus rapidement qu'en consultant le manuel.

Une seule contrainte, mais impérative : les programmes fournis doivent être scrupuleusement frappés et exécutés.

C'est en la voyant à l'oeuvre que l'on comprend le fonctionnement d'une instruction, même simple. Parcourir le listing, même avec application, ne suffit pas, car certaines réactions de la machine se trouvent escamotées.

Comment deviner, par exemple, que le curseur d'une instruction INPUT apparaît entre guillemets lorsqu'une chaîne est attendue, mais tout seul s'il s'agit d'une valeur numérique ? PASSEPORT POUR ZX81 est un bon outil de travail, mais seulement un outil. Il appartient à son lecteur de s'en servir avec application pour en tirer le maximum de profit.

P. G.

## CLEFS POUR LE ZX81 ET TIMEX 1000

Par Jean-François Sehan  
Editions du PSI  
Prix : 82 F

Il suffisait d'y penser ! Telle est en effet la conclusion qui se présente immédiatement à l'esprit au terme de la lecture de cet ouvrage "pas comme les autres".

Aide-mémoire, memento, bloc-notes, ne sont que quelques exemples de termes pouvant servir à définir ce petit livre dont même la reliure à anneaux semble inviter à une consultation de tous les instants.

On ne trouvera guère d'inédit dans CLEFS POUR LE ZX81, qui se présente un peu comme la "mise au propre" de toute la documentation qu'un passionné du ZX aurait pu glaner ici ou là en un an ou deux de recherches. Certaines rubriques proviennent en droite ligne du manuel SINCLAIR (jeu de caractères, variables-système, index des mots-clé, codes d'erreur, etc.), mais J.F. Sehan a toujours tenu à les compléter par des commentaires d'inspiration personnelle parfois fort judicieux.

Le langage machine Z80 occupe un chapitre entier, avec divers tableaux permettant l'assemblage ou le désassemblage à la main dans les meilleures conditions. On pourra regretter l'utilisation exclusive de la notation hexadécimale, mais fort heureusement un très pratique tableau de conversion permet un retour immédiat au code décimal, seul compris par le petit Sinclair.

On appréciera à sa juste valeur le recueil d'adresses commentées des routines de la ROM, qui ne pouvait guère être consulté, jusqu'à présent, que dans le rare et coûteux livret de Ian Logan, dont l'auteur a fort probablement pris le risque de s'inspirer très largement...

Après le logiciel, le matériel : le schéma du ZX81 n'étant fourni qu'aux acheteurs de la version "kit", les autres seront heureux de trouver un plan simplifié (peut-être un peu trop !), et des informations détaillées sur les brochages des circuits intégrés et du connecteur arrière.

Bien qu'utiles, ces données techniques ne suffiront pas aux risque-tout désireux d'intervenir à l'intérieur même de la machine, et qui devront se procurer



ailleurs le schéma complet, qui aurait indiscutablement été à sa place parmi cette abondance de documentation.

En guise de conclusion, un chapitre "TRCUS ET ASTUCES" fournit une sélection de tours de main qui nous rappellent pour la plupart des lectures antérieures...

Les grandes qualités de ce livre sont celles de la collection où il figure. Le mérite personnel de J.F. Sehan est d'avoir regroupé de façon cohérente un grand nombre d'informations puisées aux sources les plus diverses : ce petit livre de cent cinquante grammes en dit autant sur le ZX81 qu'une quinzaine de kilos de documentation brute.

Beau travail de synthèse donc, malgré une nette faiblesse du côté des données relatives au circuit interne de la machine. Quoi qu'il en soit, CLEFS POUR LE ZX81 rendra de fiers services à l'utilisateur "sérieux" du ZX81, en lui évitant de perdre un temps précieux à compulser pour un point précis une documentation personnelle plus éparpillée.

P. G.

#### MATHEMATIQUES SUR ZX81

Par Michel Rousselet  
Collection Poche Informatique  
Prix : 50 F environ  
128 pages

Les Fonctions mathématiques du ZX81, très riches, ne servent le plus souvent qu'à la construction de décors pour les jeux vidéo les plus divers. Cela est bien dommage, car une telle puissance de calcul aurait fait la joie des étudiants des années soixante-dix, peinant sur d'antiques engins à réglette et curseur coulissants...

Il n'est cependant pas trop tard pour réagir, comme en témoigne cet intéressant petit livre, qui s'adresse sans nul doute à un public étudiant ou scientifique. Dès le troisième chapitre, le niveau du discours rejoint celui de la classe de "Math'sup" pour dépasser allègrement celui de "Math'spé" vers la fin du volume !

Nous étions bien loin de soupçonner, avant de lire cet ouvrage, que le ZX81 pouvait résoudre des équations différentielles, intégrer des fonctions numériques par les méthodes de Simpson et Tchebychev (pas moins...), et faire merveille en matière de probabilités et statistiques.

Tout cela non pas par simple amour de l'Art, car les programmes proposés peuvent rendre de fiers services dans bien des disciplines scientifiques, même de haut niveau, soit directement, soit par incorporation dans des logiciels plus importants.

Ce livre original est un cadeau absolument idéal (et peu coûteux, puisqu'édité en collection de poche) à faire à un lycéen enclin à passer un peu trop de temps devant son clavier, au point parfois de négliger ses révisions...

Bien que l'usage d'un ZX81 ne soit pas encore autorisé en salle d'examen, gageons que le temps passé devant un écran sur lequel se résout une équation diofantienne (mais oui !), sera plus profitable qu'une partie d'astéroïdes ou d'envahisseurs.

L'informatique individuelle risque de décevoir certains de ses adeptes par manque d'applications pratiques. MATHEMATIQUES SUR ZX81 constitue un remède efficace contre ce mal naissant, applicable d'ailleurs à d'autres machines programmables en BASIC, et notamment au SPECTRUM.

P. G.

SPECTRUM HARDWARE MANUAL  
Par Adrian Dickens  
Editeur : MELBOURNE HOUSE  
Prix : 120 F environ  
150 pages

Voici un livre qui s'adresse particulièrement aux électroniciens. Il contient, en effet, le schéma complet du ZX Spectrum, la description technique des composants et des projets de périphériques.

Dans les premiers chapitres, l'auteur passe en revue tous les composants discrets (alimentation, circuit vidéo, circuit H.P.) et monolithiques (microprocesseur, mémoire morte, etc...), en précisant la fonction de tous les signaux présents à chaque broche des boîtiers. Nous apprenons ainsi que l'alimentation d'un Spectrum 16 K peut supporter d'alimenter des circuits extérieurs jusqu'à concurrence de 300 mA, mais qu'un Spectrum de 48 K ne le peut pas. On nous explique également que la mémoire vive de base de 16 K peut être employée par le microprocesseur pour dérouler un programme BASIC, ou pour manipuler les variables système, et également, par le circuit spécialisé Sinclair C.I. 1 pour réaliser l'affichage. Cette demande de mémoire par 2 circuits différents crée un conflit que le circuit spécialisé Sinclair règle en suspendant l'horloge du microprocesseur pendant quelques cycles. On nous dit que les signaux vidéo n'ont pas été raccordés au connecteur arrière. On nous explique les différences entre les différentes versions du ZX Spectrum, et beaucoup d'autres détails. Chaque chapitre débute par la photographie (vue côté composants) du Spectrum, version 1 et version 2, avec encadrement des éléments analysés. Cela fait un bon nombre de pages identiques, d'autant plus inutiles qu'en appendice on trouve l'emplacement des composants pour chaque version. Nous avons trouvé insuffisants les renseignements nécessaires pour transformer un Spectrum 16 K en un 48 K ; des éléments pour le réaliser sont cependant présents dans le livre.

Dans les derniers chapitres (un gros tiers de l'ouvrage), l'auteur donne des éléments pour l'addition de périphériques, par l'intermédiaire du connecteur arrière. Après avoir expérimenté quelques circuits simples (entre autres, un bouton RESET, bien utile quand l'ordinateur se plante : on n'a plus besoin de couper puis de reconnecter l'alim.), l'auteur décrit quelques réalisations : un PIO qui est une interface parallèle d'E/S avec lequel on dispose de 2 ports de 8 bits chacun ; une interface pour raccorder un 2ème clavier alphanumérique ou un clavier hexadécimal ou tout arrangement de touches souhaité ; un ou deux joysticks pour la commande de mouvements avec un programme Apollo comme exemple et enfin, un convertisseur analogique/digital avec un programme pour dessiner sur l'écran comme exemple d'utilisation. Le fait que ces réalisations sont accompagnées de schémas de principe, de schémas de câblage et de nomenclatures les rendent accessibles aux bricoleurs éclairés, cible directe de ce livre soigné.

Y. D.

## PROGRAMMER EN FORTH

Par Alain Pinaud  
Editions du PSI  
Prix : 82 F environ  
160 pages

Le FORTH est en train de susciter un intérêt croissant et ORDI-5 a des raisons particulières de s'intéresser à lui : c'est en effet la langue "vernaculaire" d'un ordinateur très sympathique que nous avons décidé d'adopter dans notre revue, le Jupiter Ace, Désormais tous les ordinateurs Sinclair peuvent aussi être implementés en FORTH grâce à des cassettes supportant ce langage : il y a même désormais plusieurs versions par ordinateur, produite chacune par une maison de soft réputée outre manche.

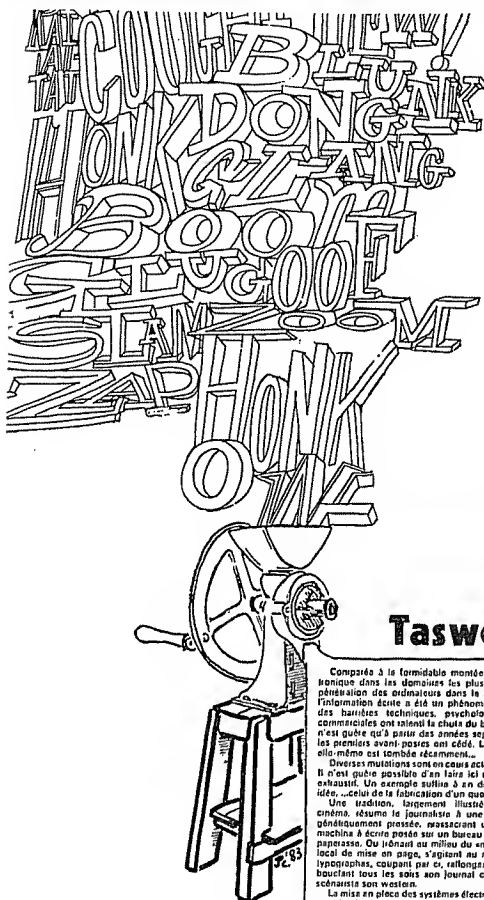
Seulement voilà : existe-t-il un livre simple pour apprendre ce langage somme toute difficile ? Tant qu'il était nécessaire de se tourner vers la documentation anglo-américaine pour approfondir ses connaissances dans ce domaine la réponse était "non" ; depuis la parution chez PSI du livre de Alain Pinaud : "programmer en forth" la réponse est péremptoirement "oui".

Chaque auteur a son truc : celui d'Alain Pinaud est de savoir présenter des choses difficiles dans un style enjoué et qui passe très bien ; nous avons déjà beaucoup apprécié son livre "programmer en assembleur" grâce auquel plus d'un possesseur de ZX81 a osé

s'aventurer dans le domaine du microprocesseur Z80 ; pourtant l'assembleur Z80 n'est pas chose aisée à présenter : on compte sur les doigts les bons livres qui peuvent en parler simplement. Les mêmes qualités se retrouvent dans cet ouvrage : l'humour le dispute à la compétence. L'approche est progressive et assez gratifiante à condition d'avoir son Jupiter Ace ou son TRS80 sous la main constamment, et de bons résumés font le point régulièrement, sans négliger les exercices qui constituent l'un des points forts de la collection.

Pour tous ceux qui possèdent le Jupiter Ace et qui trouvent que le manuel qui l'accompagne pêche souvent par éllision, un second livre est absolument nécessaire et nous conseillons celui-là. En termes d'initiation, cet ouvrage est pratiquement sans défauts. Une chose manque cependant absolument et nous voudrions bien que dans une seconde édition il y soit porté remède : l'absence d'un index analytique se fait cruellement sentir et rend le livre, une fois lu, difficile à manier et remanier au titre de la référence. Mais il est très probable que le "fortheur" qui aura traversé ce livre avec profit envisagera l'acquisition d'un autre ouvrage plus théorique et plus systématique. Cet ouvrage pourrait s'intituler très justement le "forth sans peine" si la vénérable collection que chacun connaît s'ouvrait un jour aux langages de programmation.

X. L. B.



SEMAPHORE.

## TASWORD DEUX

LE TRAITEMENT DE TEXTE PROFESSIONNEL POUR SPECTRUM 48K.

Adaptation française de Tasword Two maintenant disponible vous offre :

- caractères accentués, cédilles, tréma etc...
- 64 caractères par ligne sur l'écran et l'imprimante (idéal pour A4),
- rapidité (écrit en code machine) mais personnalisable,
- toutes les fonctions habituelles : justification commutable, insertions, déplacement de lignes et paragraphes, formatage et margination, remplacement de mots ou groupes de mots, etc...
- manuel complet et instructions d'écran en français, (manuel aussi livrable en allemand et néerlandais)
- programme d'apprentissage et d'entraînement Tasword Tuteur,
- utilise sans interface l'imprimante ZX,
- compatible avec la majorité des interfaces et imprimantes professionnelles (Cobra RS 232 i/o port, Euroelectronics ZX Print, Kempston, Hildenbay, Morex, Tasman, etc... - Epson, Seikosha, Oki, Tandy, etc...)

Nous recommandons et fournissons l'interface Tasman centronie, programmée par Tasword Deux pour l'Epson FX-80 et livrée avec programmes sur cassette pour les Star et Tandy graphique couleur.

Une notice détaillée permet l'adaptation par l'utilisateur.

**Tasword Deux 60f.s. Interface 150f.s. Les 2: 200f.s.**

VENTE PAR CORRESPONDANCE : envoyer à SEMAPHORE logiciels,  
Jean-Pierre Cardinaux  
CH 1283 La Plaine  
(Genève) Suisse.

Je commande : Tasword deux

Tasman interface

Je verse le montant de Fs. au CCP (ou chèque) 12.247 98  
de Jean-Pierre Cardinaux à Genève.

Je verse au compte CO-192.930.1 à la Société de Banque  
Suisse à Genève.

Contre remboursement (taxes en plus)

Nom : prénom :

adresse :

Téléphone :

Comparés à la formidable montée de l'électronique dans les domaines les plus divers, la pénétration des ordinateurs dans le monde de l'information écrite a été un phénomène tardif : des barrières techniques, psychologiques et commerciales ont retardé la chute du bétail. Ce n'est qu'en 1964 que des ordinateurs ont permis les premiers avants pas vers une écriture. La cité de l'écrit même est tombée récemment... Diverses mutations sont en cours actuellement. Il n'est guère possible d'en faire ici un exposé exhaustif. Un exemple suffit à en donner une idée : celui de la fabrication d'un quotidien. Une tradition, largement illustrée par la presse, résume le journaliste à une personne endossant une machine à écrire posée sur un bureau envahi de papiers. Ou, s'agissant au milieu du siècle, le local de mise en page, s'agissant au milieu des typographes, coupant par ci, rallongeant par là, bécotant tous les mots son journal comme un écrivain son western. La mise en place des systèmes électroniques a tout bouleversé. Le journaliste écrit son texte sur un terminal, lui son choix de dépêches par le même mode. Réalise lui-même la mise en page à l'aide d'un écran graphique. Dans ce qui reste du maître, une photocomposée débite des pages pratiquement terminées ou seules manquent les illustrations.

# Trucs à Brac

## LE COIN DES INDISCRETS (POUR APPRENDRE SE MEFIER DES TRUCS DES PIRATES)

Il est souvent intéressant de LISTER un programme pour en comprendre le fonctionnement, les diverses astuces et trucs, ou encore tout simplement pour le modifier ; malheureusement la plupart des programmes vendus dans le commerce sont en langage machine (ce qui empêche l'utilisateur d'arrêter le programme en agissant sur la commande BREAK) et s'exécutent automatiquement après leur chargement par la commande LOAD "nom du programme".

Voici une petite routine toute simple, à entrer directement au clavier, qui permet de pallier à cet inconvénient et qui peut encore rendre d'énormes services (à condition de ne pas l'utiliser pour une duplication abusive de logiciels), par exemple : la traduction française de jeu en version anglaise.

### Déroulement :

- 1) Positionner le micro-ordinateur en mode FAST ;
- 2) Taper RAND USR 836 suivi de NEWLINE ;
- 3) Appuyer sur PLAY comme lors d'un LOAD normal ;
- 4) Attendre la fin du chargement : un compte-rendu du style C/... apparaît, vous pouvez LISTER votre programme.

Voilà, vous avez vu ce n'est vraiment pas compliqué, et ça rend souvent d'énormes services (par exemple : obtenir le LISTING d'un programme qui se charge automatiquement au-dessus de RAMTOP).

N.B. Une petite explication tout de même concernant le RAND USR 836, il faut savoir que dans la ROM Basic de Sinclair, il existe un micro-programme correspondant à la commande LOAD à l'adresse décimale 832 (en hexadécima : 3C2), c'est donc une partie de cette routine que l'on utilise.

Philippe Buschini

II - Entrer et faire tourner le programme I puis effacer les lignes 10 à 50.

Taper ensuite les lignes 10 et 15 du programme II que l'on sauvegarde en faisant GOTO 10. C'est le programme qu'il faudra introduire dans le ZX au lieu de taper LOAD"" quand on voudra lister un programme protégé.

```
101 REM 12345678
102 LET A=16514
103 POKE A+1,255
104 POKE A+2,125
105 POKE A+3,125
106 POKE A+4,125
107 POKE A+5,125
108 POKE A+6,125
109 POKE A+7,125
110 POKE A+8,125
```

```
1 REM LN 77ACS IF 77
10 SAVE "PIRATAGE"
15 PRINT "FAITES ""RAND USR
16514"" PUIS METTEZ VOTRE MAG
NETOPHONE EN POSITION ""LECTURE
```

François Hache

## TRUC A BREAK

Comment faire en sorte que la touche BREAK soit désensibilisée de façon à interdire le piratage de ses programmes Spectrum ? Il suffit de faire POKE 23659,0.

### SAVE LOAD (ZX81), UNE AMELIORATION DECISIVE !

La solution que j'apporte n'a, à ma connaissance, jamais été publiée malgré les nombreux articles sur le sujet.

Elle est pourtant d'une grande simplicité. Il faut intercaler, dans la liaison MIC, une résistance dont la valeur optimum est à déterminer expérimentalement et dépend essentiellement de la bonne entente des deux protagonistes.

Deux exemples précis permettent de mieux apprécier :  
- 1er cas : impossibilité absolue de réaliser un enregistrement fiable.

A la lecture, avant même l'arrivée des signaux caractéristiques, l'écran devient blanc, le ZX renvoie le curseur K.

Le ZX étant hors de cause, la partie lecture également (aucun problème pour lire une cassette pré-enregistrée pro. ou amateur) c'est bien la partie enregistrement qui est en cause. L'insertion d'une résistance de 6800 Ohms (la tolérance est assez grande) permet de réaliser des enregistrements fiables, à 90 %, en vitesse normale.

Par contre, dans ce cas-là, aucun succès avec le FAST LOAD MONITOR (FLM).

D'ailleurs, la lecture est, à ces vitesses, en cause car il est impossible de lire une cassette pré-enregistrée ; je n'ai pour l'instant aucune solution à proposer, les réglages d'Azimut sont sans effet.

- 2ème cas : aucun problème d'enregistrement à vitesse normale. Fiabilité 100 %.

Par contre en FLM aucun enregistrement n'est lisible ; la fonction 3 (titre) donne des résultats étranges : les titres affichés ressemblent au REM du code machine, ce qui me guide sur la bonne voie, le début de l'enregistrement est saturé, déformé.

L'insertion d'un potentiomètre de précision de 20 kohm me permet de constater la progression des résultats. L'augmentation progressive de la résistance série dans la liaison MIC permet de voir les titres se transformer d'abord en "AST LOAD MONITOR" puis en titres réels à la vitesse 9.

En descendant l'échelle des vitesses, on constate à nouveau un déraillement dans la reproduction du titre : augmentation de la résistance et de nouveau de bons titres.

### MODIFICATION DE RAMTOP PAR UN PROGRAMME

Bien des programmes, et notamment des jeux du commerce, nécessitent la réservation de place mémoire au-dessus de RAMTOP.

Il est alors d'usage que la notice invite l'utilisateur à exécuter un ou deux POKE en mode commande, puis un NEW. Ces préparatifs fastidieux et générateurs de



Le LOAD de la ligne 40 exécutera l'équivalent d'un NEW et chargera immédiatement le programme utile. Le SAVE de la ligne 10 permet d'enregistrer le programme amorcé par un simple RUN, tout en lui permettant de se lancer seul lors du rechargement. Ce procédé est ainsi entièrement "transparent" à l'utilisateur qui n'aura aucune consigne particulière à suivre pour charger la cassette par un simple LOAD". Il sera avantageux d'enregistrer le programme amorcé sans extension mémoire en place : on accélérera ainsi risques d'erreurs peuvent être facilement éliminés par un chargement en deux temps du logiciel. Il n'est pas possible, en effet, de placer ces instructions en tête du programme principal, puisque le NEW nécessaire effacerait toute la partie utile du logiciel !

On peut par contre loger sur la cassette, juste avant le programme principal, les quelques lignes suivantes :

```
10 SAVE "AMORCE"
20 POKE 16388, octet bas de Ramtop
30 POKE 16389, octet haut de Ramtop
40 LOAD ""
```

notamment les opérations.

Enfin, on peut songer à ajouter les lignes suivantes, qui permettent un rapide contrôle du réglage du magnétophone :

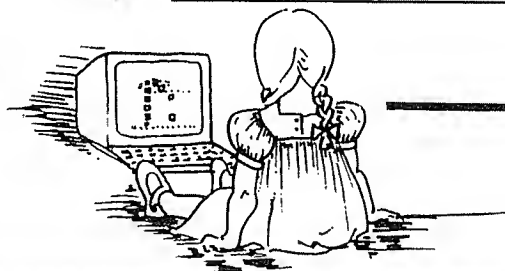
```
32 PRINT AT 10,0; "NIVEAU CORRECT"
34 PRINT AT 15,0; "ATTENDEZ SVP"
36 PAUSE 200
```

Patrick Gueulle

## pour vos enfants

# Juvenilia

Michel Murbach



## Le Schtroumpf programmeur

18 - ORDI-5 N° 6 - Décembre 1983

Qu'est-ce que la rubrique "Juvenilia" ?

C'est le point de rencontre des oeuvres de jeunesse, comme disent les latinistes en parlant des grands auteurs classiques. La rédaction d'ORDI-5 a dans l'idée que sur les quelque 50 000 utilisateurs francophones du ZX81, il doit y avoir une très forte proportion de jeunes et même de très jeunes.

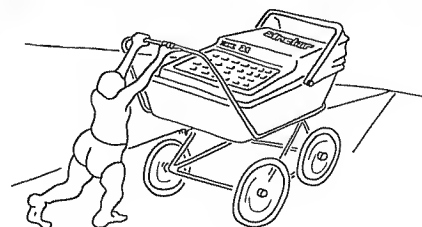
Vous qui avez moins de 14 ans, cette rubrique vous est ouverte. Bien sûr, si vous êtes déjà capable, comme le jeune Simon LANE, de nous écrire un programme de 800 instructions pour la résolution du Rubik's Cube, on vous trouvera une autre place dans la revue !

Juvenilia 1K

```
1 LET N=CODE " "
2 LET D=CODE " "
3 INPUT H
4 LET Q=D
5 LET X=N
6 LET Y=N
7 LET T=X+CODE " "
8 GOTO 55
9 GOTO 55
10 PRINT AT N,X; "*" ; TAB X+H; "*"
11
12 LET T=T+CODE " "
13 LET D=D+(AND-RND)/CODE " "
14 IF ABS D>X/X THEN LET D=X-X
15 LET X=X+SGN D
16 IF NOT X THEN LET X=Y/Y
17 IF X>CODE " " THEN LET X=20
18 SCROLL
19 PRINT AT CODE " ",Y-1; " "
20 LET A$=INKEY$
21 IF X>CODE " " AND X<>CODE " "
22 THEN LET K=K+SGN D
23 IF A$="S" THEN LET Y=Y-X/X
24 IF A$="8" THEN LET Y=Y+X/X
25 IF T/N=INT (T/N) AND (Y<R+K
26 OR Y>R+K) THEN GOTO 90
27 IF T/N=INT (T/N) THEN LET K
28 =CODE " "
29 IF T/N=INT (T/N) THEN LET R
30 =X
31 GOTO CODE " "
32 PRINT T,Q
33 IF T>0 THEN LET Q=T
34 GOTO CODE " "
```

Bruno Célerier

## Rallye



## Zedix ou l'ordinateur expliqué par lui-même

Ce beau programme éducatif de Claude Remy affiche un éclaté du ZX avec ses différents circuits intégrés et les liaisons qui les relient entre eux et simule d'une façon assez spectaculaire le cheminement des impulsions de courant et les modifications d'état des cellules de mémoire. Après cela le ZX ne vous paraîtra pas plus compliqué qu'un beau boulier, même si vous avez un peu tâtonné au début.

### LA PLANCHE D'ANATOMIE DU ZX : LA MEMOIRE, L'UNITE DE TRAITEMENT ET LES LIAISONS

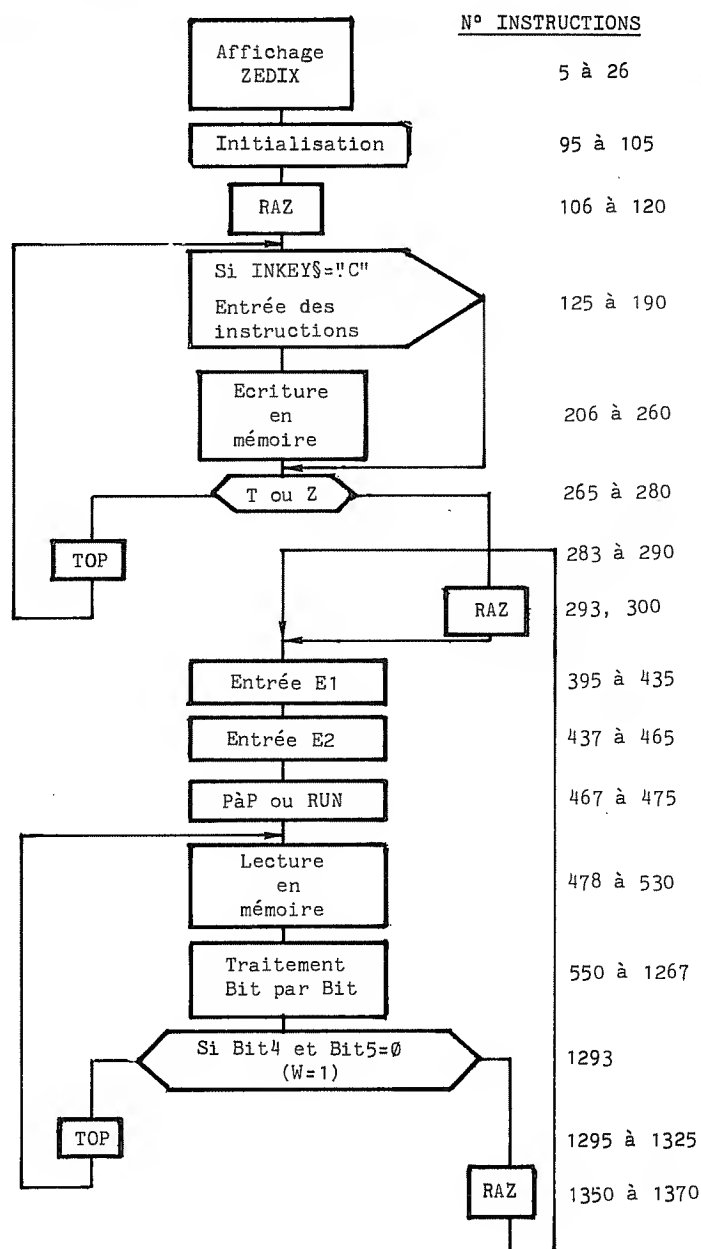
Le ZX est un ordinateur très simple mais qui possède les organes essentiels des grands systèmes : une mémoire et une unité de traitement.  
Comme tout ordinateur, il ne connaît que 2 états :

- le courant électrique passe : bit 1 : XXXXXXXXXX
- le courant ne passe pas : bit 0 : .....

\* La mémoire permet de stocker 64 bits ou 8 mots de 8 bits. Chaque mot occupe une rangée. Chaque emplacement de bit est représenté par un point.  
Les mots entrent ou sortent de la mémoire par 8 fils raccordés au bus dont les 8 fils ont été réunis en un gros trait.  
Pour accéder à une rangée (pour y écrire ou pour lire un mot) il faut que la ligne-mémoire correspondante, qui sort du décodeur, soit à l'état 1.  
Les lignes-mémoire sont successivement à l'état 1 au rythme des tops délivrés par une horloge (non représentée sur l'écran).  
Les tops (bref passage de courant électrique) sont comptés par un compteur qui délivre un nombre binaire correspondant au nombre de tops.  
Ce nombre binaire constitue l'adresse de la mémoire. Elle est décodée par le décodeur qui, à chaque nombre binaire, fait correspondre dans l'ordre, une ligne-mémoire.  
Pour comprendre le principe de représentation à l'écran :

- . Initialiser le système par une remise à zéro (Raz : Touche Z)
- . Envoyer successivement des tops (Touche T) - (taper C (comme Compteur) quand l'écran demande 0 ou 1).

	Compteur		Ligne-Mémoire
	Système décimal	Nb binaire = adresse	
Raz	0	000	1
1er Top	1	001	2
2e Top	2	010	3
3e Top	3	011	4
...	ETC ...		
7e Top	7	111	8



\* L'unité de traitement n'opère que sur des entrées à un bit (0 ou 1).  
Les données sont introduites en E1 et E2.  
Le traitement consiste à résoudre les fonctions logiques de base : ET, OU, PAS :

- Fonction ET : le résultat = 1 si les 2 entrées sont égales à 1.
- Fonction OU : le résultat = 1 si une entrée au moins est égale à 1.
- Fonction PAS : (n'opère que sur une seule entrée) : le résultat = 1 si l'entrée est égale à 0.

L'opération logique est effectuée par l'unité de calcul LOG.

Le résultat apparaît dans les cases R1 ou R2. Ce résultat peut être réintroduit par les entrées E3 et E4. Chaque entrée est validée par une porte ET "\*", qui laisse passer le bit de l'entrée si la ligne venant du bus est à l'état 1. Il en est de même pour les résultats issus de LOG.

L'opération effectuée par l'unité de calcul LOG dépend de l'état des lignes 4 et 5 issues du bus, selon le code suivant :

Ligne 4	Ligne 5	Opération effectuée
0	0	Arrêt
0	1	Opération ET
1	0	Opération OU
1	1	Opération PAS

On peut maintenant déterminer les instructions à écrire en mémoire et qui commanderont l'unité de traitement :

1er exemple :

Opération ET entre E1 et E2, résultat en R1

	Validation des entrées				Code opératoire		Validation de la sortie		
N° de la ligne de bus	0	1	2	3	4	5	6	7	
E1 ET E2 = R1	1	1	0	0	0	1	0	1	

2ème exemple :

Opération PAS sur l'entrée E3, résultat en R2  
PAS E3 = R2 : 00101101 ou 2D en hexadécimal

Il faut commencer par stocker les instructions en mémoire.

Pour écrire un mot en mémoire, il faut :

- qu'un mot soit présent sur le bus. Les mots-instructions sont introduits par l'organe d'entrée 0 à 7 à gauche du bus.
- que la ligne-mémoire correspondant à la rangée désirée, soit à l'état 1 et par conséquent qu'une adresse soit appliquée au décodeur.
- que l'ordre E (écriture) soit à l'état 1.

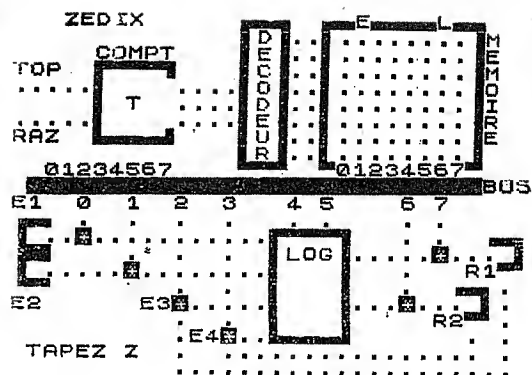
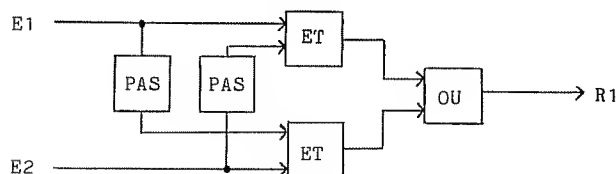
De même, pour lire un mot en mémoire, il faudra :

- que la ligne-mémoire correspondante soit à l'état 1.
- que l'ordre L (lecture) soit à l'état 1.

#### VISUALISEZ UN EXEMPLE DE TRAITEMENT

Essayons de résoudre la fonction OUX (OU exclusif) :

R1 = 1 si une seule des entrées E1 ou E2 = 1



```

1 REM *ZEDIX*
2 REM (C) CL. REHY ET ORDI-5
3 PRINT "----- ZEDIX -----"
4
5 PRINT TR8 14; "D1....."
6
7 PRINT "      COMPT  E1..E
8 PRINT "TOP      IC1..E
9 PRINT "      ..... 101..E
10 PRINT "      T      D1..E
11 PRINT "      ..... E1..E
12 PRINT "RAZ      IU..E
13 PRINT TAB 14; "R1....."
14 PRINT "      01234567
15 PRINT "      BUS"
16 PRINT "E1 0 1 2 3 4 5
17 PRINT "
18 PRINT "      ..... LOG
19 PRINT "      R1
20 PRINT "      E2      E3.....
21 PRINT "      TR8 10; ".
22 PRINT "      E4.....
23 PRINT TAB 10; ".
24 PRINT TAB 10; ".
25 PRINT TAB 10; ".
26 PRINT TAB 10; ".
27
28 DIM H(8,8)
29 LET K=0
30 LET L=0
31 PRINT RT 20,1;"TAPEZ Z"
32 IF INKEY#="Z" THEN GOTO 110
33 GOTO 110
34 GOSUB 9000
35 GOSUB 6000
36 PRINT RT 20,1;"0 OU 1 ?"
37 LET C=0
38 LET L=L+1
39 IF INKEY#="" THEN GOTO 130
40 IF INKEY#=" " THEN GOTO 135
41 LET H#:=INKEY#
42 IF H#="C" THEN GOTO 265
43 LET C=C+1
44 IF C=8 THEN GOTO 130
45 PRINT RT 21,C;H(L,C)
46 IF H#="0" THEN PRINT RT 9,(
47 C+1);CHR$(27+C);TAB (C+19);CHR$
48 (27+C)
49 IF H#="1" THEN PRINT RT 9,(
50 C+1);CHR$(155+C);TAB (C+19);CHR$
51 (155+C)
52 IF C=8 THEN GOTO 206
53 GOTO 130
54 PRINT RT 21,1;"
55 PRINT RT 20,1;"TAPEZ E "
56 IF INKEY#="E" THEN GOTO 210
57 GOTO 210
58 GOSUB 9000
59 PRINT RT 20,21;"E"
60 FOR C=1 TO B
61 IF M(L,C)=0 THEN GOTO 250
62 PRINT RT L,(19+C);" "
63 GOTO 255
64 PRINT RT L,(19+C);"."
65 NEXT C
66 PRINT RT 20,21;"E"
67 PRINT RT 20,1;"T OU Z "
68 IF INKEY#="Z" THEN GOTO 293
69 IF INKEY#="T" THEN GOTO 283
70 GOTO 270
71 GOSUB 9000
72 GOSUB 6000
73 GOTO 125
74 GOSUB 9000
75 GOSUB 5000
76 PRINT RT 9,2;"
77 LET V=0
78 LET S=0
79 LET U=0
80 PRINT RT 20,1;"E1 ? "
81 IF INKEY#="0" THEN GOTO 412
82 IF INKEY#="1" THEN GOTO 411
83 GOTO 409
84 LET D=0
85 GOTO 415
  
```



```

414 LET D=1
415 PRINT AT 13,1;D
420 IF D=0 THEN GOTO 435
420 PRINT AT 13,2;"**"
430 GOTO 437
430 PRINT AT 13,2;"E."
437 PRINT AT 20,1;"E?"
439 IF INKEY#="" THEN GOTO 442
440 IF INKEY#="1" THEN GOTO 444
441 GOTO 439
442 LET E=0
443 GOTO 445
444 LET E=1
445 PRINT AT 15,1;E
450 IF E=0 THEN GOTO 465
450 PRINT AT 15,2;"*****"
460 GOTO 467
460 PRINT AT 15,2;"P.A.P."
467 PRINT AT 19,1;"P.A.P."
468 PRINT AT 20,1;"OU RUN:R"
470 IF INKEY#="" THEN GOTO 470
471 IF INKEY#="" THEN GOTO 471
472 LET K#="INKEY#"
473 GOSUB 9000
474 IF K#="R" THEN GOTO 488
475 PRINT AT 20,1;"TAPEZ L"
476 IF INKEY#="L" THEN GOTO 483
480 GOTO 480
480 GOSUB 9000
480 PRINT AT 0,26;"E"
485 LET L=T+1
485 FOR C=1 TO 8
485 IF M(L,C)=0 THEN GOTO 520
485 PRINT AT 9,(19+C);CHR$(155)
500 GOTO 525
500 PRINT AT 9,(19+C);CHR$(27+
500)
500 NEXT C
500 PRINT AT 0,26;"L"
500 IF M(L,1)=0 THEN GOTO 585
500 PRINT AT 11,4;"0"
500 PRINT TAB 4;"#"
500 IF D=0 THEN GOTO 595
500 LET D=1
500 PRINT TAB 5;"*****"
500 GOTO 650
500 PRINT AT 11,4;"0"
500 PRINT TAB 4;"#"
500 LET N=0
500 PRINT TAB 5;"*****"
500 IF M(L,2)=0 THEN GOTO 685
500 PRINT AT 11,7;"E"
500 FOR X=2 TO 14
500 PRINT AT X,7;"#"
500 NEXT X
500 IF E=0 THEN GOTO 695
500 LET E=1
500 PRINT AT 15,8;"*****"
500 GOTO 750
500 PRINT AT 11,7;"1"
500 PRINT AT 12,7;"2"
500 PRINT AT 14,7;"3"
500 LET O=0
500 PRINT AT 15,8;"*****"
500 IF M(L,3)=0 THEN GOTO 785
500 PRINT AT 11,10;"E"
500 FOR X=12 TO 16
500 PRINT AT X,10;"#"
500 NEXT X
500 IF P=0 THEN GOTO 795
500 LET P=1
500 PRINT AT 17,11;"*****"
500 GOTO 850
500 PRINT AT 11,10;"2"
500 FOR X=12 TO 16 STEP 2
500 PRINT AT X,10;"#"
500 NEXT X
500 LET P=0
500 PRINT AT 17,11;"*****"
500 IF M(L,4)=0 THEN GOTO 885
500 PRINT AT 11,13;"E"
500 FOR X=12 TO 16
500 PRINT AT X,13;"#"
500 NEXT X
500 IF S=0 THEN GOTO 895
500 LET S=1
500 PRINT AT 19,14;"**"
500 GOTO 950
500 PRINT AT 11,13;"3"
500 FOR X=12 TO 16 STEP 2
500 PRINT AT X,13;"#"
500 NEXT X
500 LET S=0
500 PRINT AT 19,14;"**"
500 IF M(L,5)=0 THEN GOTO 1038
500 PRINT AT 11,17;"E"
500 PRINT TAB 4;"#"
500 IF M(L,6)=0 THEN GOTO 1030
500 PRINT AT 11,19;"E"
500 PRINT AT 12,19;"2"
500 PRINT AT 16,17;"PAS"
500 LET U=1
500 IF N=1 OR O=1 OR P=1 OR S=1
500 THEN LET U=0
500 GOTO 1055
500 PRINT AT 11,19;"5"
500 PRINT TAB 19;"#"
500 PRINT AT 16,17;"OU"
500 LET U=0
500 IF N=1 OR O=1 OR P=1 OR S=1
500 THEN LET U=1
500 GOTO 1055
500 PRINT AT 11,17;"4"
500 PRINT AT 12,17;"2"
500 IF M(L,8)=0 THEN GOTO 1088
500 PRINT AT 11,19;"E"
500 PRINT AT 12,19;"2"
500 PRINT AT 16,17;"ET"
500 LET U=0
500 IF (N=1 AND O=1) OR (N=1 AND
500 P=1) OR (N=1 AND S=1) OR (O=1
500 AND P=1) OR (O=1 AND S=1) OR (P=
500 AND S=1) THEN LET U=1
500 IF U=0 THEN GOTO 1075
500 PRINT AT 14,21;"*****"
500 PRINT AT 17,21;"*****"
500 GOTO 1100
500 PRINT AT 14,21;"*****"
500 PRINT AT 17,21;"*****"
500 GOTO 1100
500 PRINT AT 11,19;"5"
500 PRINT AT 12,19;"2"
500 LET U=1
500 PRINT AT 16,17;"FIN"

```

```

1100 IF M(L,7)=0 THEN GOTO 1135
1105 PRINT AT 11,24;"E"
1110 FOR X=12 TO 16
1110 PRINT AT X,24;"#"
1112 NEXT X
1113 IF U=0 THEN GOTO 1148
1113 LET S=1
1113 PRINT AT 17,25;"***"
1113 PRINT AT 17,28;"S"
1113 PRINT AT 19,28;"#"
1113 PRINT AT 20,13;"*****"
1134 GOTO 1200
1135 PRINT AT 11,24;"6"
1140 FOR X=12 TO 16
1140 PRINT AT X,24;"#"
1141 IF X=14 AND U=1 THEN GOTO 1
143
1142 PRINT AT X,24;"#"
1143 NEXT X
1144 PRINT AT 17,25;"***"
1145 GOTO 1200
1145 LET S=0
1145 PRINT AT 17,25;"***"
1145 PRINT AT 17,28;"S"
1145 PRINT AT 19,28;"#"
1145 PRINT AT 20,13;"*****"
1200 IF M(L,6)=0 THEN GOTO 1235
1205 PRINT AT 11,26;"E"
1210 PRINT AT 12,26;"2"
1211 PRINT AT 13,26;"#"
1211 IF U=0 THEN GOTO 1245
1220 LET U=1
1220 PRINT AT 14,27;"***"
1224 PRINT AT 14,30;"U"
1225 FOR X=16 TO 21
1225 PRINT AT X,30;"#"
1227 NEXT X
1228 FOR Y=30 TO 10 STEP -1
1228 PRINT AT 21,Y;"#"
1230 NEXT Y
1230 FOR X=21 TO 16 STEP -1
1230 PRINT AT X,10;"#"
1233 NEXT X
1234 GOTO 1293
1235 PRINT AT 11,26;"7"
1240 PRINT AT 12,26;"2"
1241 PRINT AT 13,26;"#"
1242 PRINT AT 14,27;"***"
1243 GOTO 1293
1250 LET U=0
1250 PRINT AT 14,27;"***"
1252 PRINT AT 14,30;"U"
1255 FOR X=16 TO 21
1255 PRINT AT X,30;"#"
1257 NEXT X
1258 FOR Y=30 TO 10 STEP -1
1258 PRINT AT 21,Y;"#"
1260 NEXT Y
1260 FOR X=21 TO 16 STEP -1
1260 PRINT AT X,10;"#"
1263 NEXT X
1263 IF U=1 THEN GOTO 1350
1265 IF K#="R" THEN GOTO 1320
1300 PRINT AT 20,1;"TAPEZ T"
1310 IF INKEY#="T" THEN GOTO 131
1315 GOTO 1310
1315 GOSUB 9000
1320 GOSUB 9000
1325 GOTO 475
1350 PRINT AT 20,1;"TAPEZ Z"
1355 IF INKEY#="Z" THEN GOTO 135
1360 GOTO 1355
1365 GOSUB 9000
1365 GOSUB 9000
1370 GOTO 435
5000 LET T=0
5005 FOR N=0 TO 4
5010 PRINT AT 6,N;"#"
5015 PRINT AT 6,(N+1);"#"
5020 PRINT AT 6,N;"#"
5025 NEXT N
5025 GOSUB 9000
5030 PRINT AT 4,5;"E"
5033 GOSUB 9000
5035 IF T=1 OR T=3 OR T=5 OR T=7
5035 THEN LET A#="1"
5040 IF T=2 OR T=3 OR T=6 OR T=7
5040 THEN LET B#="1"
5045 IF T=4 OR T=5 OR T=6 OR T=7
5045 THEN LET C#="1"
5050 GOSUB 7000
5055 LET K=T+1
5060 RETURN
7000 PRINT AT 5,7;T
7005 PRINT AT 4,9;A#
7010 PRINT TAB 9;"#"
7015 PRINT TAB 9;"#"
7020 IF A#="1" THEN PRINT AT 4,1
7025 IF A#="0" THEN PRINT AT 4,1
7030 IF B#="1" THEN PRINT AT 5,1
7035 IF B#="0" THEN PRINT AT 5,1
7040 IF C#="1" THEN PRINT AT 6,1
7045 IF C#="0" THEN PRINT AT 6,1
7050 PRINT AT (T+1),17;"***"
7055 IF T>0 THEN PRINT AT T,17;"
7060 RETURN
7065 LET A#="0"
7065 LET B#="0"
7070 LET C#="0"
7075 RETURN
7080 PRINT AT 19,1;" "
7085 PRINT AT 20,1;" "
7090 RETURN

```

GOSUB 5000 : Exécution RAZ  
 GOSUB 6000 : Exécution TOP  
 GOSUB 7000 : Affichage T dans le compteur  
 Affichage du nombre binaire à la sortie  
 du compteur  
 Affichage des fils adresse  
 Affichage de la ligne-mémoire  
 GOSUB 8000 : Remise à zéro de A\$, B\$, C\$  
 GOSUB 9000 : Effacement des lignes Dialogue 19 et 20.

#### \* Ecriture des instructions en mémoire

1ère instruction :

PAS E1 = R2 10001110 8Eh

Faire une remise à zéro RAZ.

Touche Z

Introduire par les touches 0 et 1 le mot ci-dessus  
 (les bits 1 apparaissent en vidéo inversée).

Envoyer l'ordre E.

Touche E

Le mot s'inscrit en mémoire, les bits 1 sont représentés par E

2ème instruction :

E2 ET E4 = R2 01010110 56h

Envoyer un TOP

Touche T

Introduire le 2ème mot

Touches 0 et 1

Envoyer l'ordre E

Touche E

Procéder de même pour les instructions suivantes.

3ème instruction : PAS E2 = R1 01001101 4Dh  
 4ème instruction : E1 ET E3 = R1 10100101 A5h  
 5ème instruction : E3 OU E4 = R1 00111001 39h  
 6ème instruction : Arrêt 00000000 00h

Faire une RAZ (touche Z) pour réinitialiser la mémoire.

#### \* Lecture des instructions et exécution

Le programme (ensemble des instructions d'un traitement) ainsi introduit, il reste à entrer les données E1 et E2 (0 ou 1) par les touches 0 et 1.

Le traitement peut être effectué suivant deux modes :

Le mode PAS A PAS pour mieux suivre le déroulement (touche P) : vous envoyez vous-mêmes les Tops T et les ordres K.

• Envoyer l'ordre lecture

"Touche L"

La 1ère instruction est lue en mémoire et, par le BUS, commande le traitement.

• Quand le résultat est affiché, envoyer le 1er Top "Touche T".

La 2ème instruction est lue et est exécutée.

Et ainsi de suite...

La dernière instruction stoppe le traitement.

Le mode RUN : le traitement s'opère en continu "Touche R".

Vous pouvez recommencer le traitement sur de nouvelles données. Pour exécuter un autre programme, faire BREAK, RUN et NEWLINE.

Claude Remy

## initiation

# Un ordinateur comment ça marche ?

L'ordinateur a envahi tous les compartiments de la vie moderne. D'abord réservé aux grosses sociétés, il s'est petit à petit, introduit partout au fur et à mesure que son prix diminuait et que sa technique s'affinait.

En lui ajoutant le son et la couleur, il est devenu micro-ordinateur familial pour lequel on assiste actuellement à une véritable explosion.

Le merveilleux ordinateur qu'est le ZX81, a été le détonateur de cette explosion. Sa réussite a donné naissance à un tas d'autres micro-ordinateurs. Depuis, Sinclair a sorti le ZX Spectrum dont les performances, du point de vue qualité/prix, n'étaient pas imaginables il y a quelques années à peine.

Les ordinateurs, et les micro-ordinateurs, sont habituellement comparés entre eux suivant la puissance de leurs jeux d'instructions, le nombre de routines dont ils disposent, les modes d'adressage qu'ils comportent, leurs étendues mémoire, leurs possibilités graphiques et leurs vitesses d'exécution. Autant de choses qui traduisent un fonctionnement interne complexe que le non-initié a du mal à comprendre. L'ambition de cette brève introduction est de clarifier un peu ces problèmes en montrant la simplicité de certains principes de base, valables depuis les origines.

DE BLAISE PASCAL A CLIVE SINCLAIR

Les principes de base d'une petite machine à calculer ont été développées par Pascal, qui avait réalisé une

additionneuse entièrement mécanique. Les principes d'une calculatrice moderne ont été mis au point par Babbage vers les années 1830. Mais c'est Von Neumann, en 1945, qui a introduit le véritable ordinateur en appliquant deux concepts nouveaux fondamentaux :

1 - Le programme enregistré : le résultat est une calculatrice programmable dont le programme est figé dans une mémoire.

2 - Le rupture de séquence : en cours d'exécution du programme, la machine peut prendre une décision si telle condition est réalisée et cela, grâce à un branchement conditionnel ou à un saut qui rompt la séquence normale du programme.

Les machines basées sur les conceptions de Von Neumann sont toujours employées actuellement bien que des conceptions nouvelles soient appliquées aux gros ordinateurs.

On distingue plusieurs générations dans l'évolution des ordinateurs :

1ère génération : vers 1950 l'ordinateur à tubes électroniques qui remplissait une pièce.

2ème génération : vers 1962 l'ordinateur à transistors qui remplissait une console.

3ème génération : vers 1970 l'ordinateur à circuits intégrés qui remplit un tiroir.

4ème génération : vers 1975 l'ordinateur à circuits L.S.I. qui remplit une carte.

En plus de la miniaturisation évidente, des progrès ont été accomplis parallèlement en fiabilité, en prix et en vitesse d'exécution. L'ordinateur a suivi l'évolution des circuits intégrés. Ceux-ci, au départ,

dans la série 7400, avaient quelques portes intégrées sur une puce de silicium. Depuis, on en a intégré des milliers. C'est d'ailleurs la dimension et non la complexité de la puce qui détermine son prix. Les différentes étapes ont été, grosso modo :

- S.S.I. (Small Scale Integration) : intégration à petite échelle comportant quelques dizaines de portes par puce,
- M.S.I. (Medium Scale Integration) : intégration à moyenne échelle comportant quelques centaines de portes par puce,
- L.S.I. (Large Scale Integration) : intégration à grande échelle comportant quelques milliers de portes par puce.

On parle maintenant d'une 5ème génération d'ordinateurs avec le V.L.S.I. (Very Large Scale Integration). Mais tout ne peut pas se miniaturiser. Il y a l'interface humain : l'homme doit pouvoir manipuler l'ordinateur en appuyant sur des touches, en tournant des clés, des combinatoires, etc...

Les dimensions des ZX81 et Spectrum sont à leur minimum possible : le clavier doit rester accessible à son utilisateur. Par contre, ce qu'il y a en dessous du clavier évoluera encore. Nous aurons des circuits plus denses, plus performants. Nous aurons plus de mémoires, plus de routines, plusieurs langages de programmation, plus d'interfaces pour imprimantes et autres périphériques, etc...

#### DANS LA BOITE

La constitution d'un ordinateur, basée sur la logique et le calcul binaire, reste fondamentalement la même, qu'il s'agisse d'un gros, d'un mini ou d'un micro-ordinateur. La distinction entre ces classes d'ordinateurs réside dans la dimension des mots traités. Les gros ordinateurs traitent des mots de 32 bits, les minis, des mots de 16 bits et les micros, de 8 bits. On parle, bien sûr, de micro-ordinateurs de 16 bits. Ce ne sont que des minis qui se miniaturisent. L'évolution de la technologie les rend compétitifs.

La figure 1 montre la constitution d'un ordinateur. L'unité 1, qu'on appelle processeur central ou unité centrale de traitement ou microprocesseur, est le cœur du système. Dans les ZX, on emploie le microprocesseur Z80 fabriqué par Zilog. C'est ce processeur qui exécute les opérations, qui range les résultats, qui décode les instructions à réaliser et qui établit la séquence du déroulement de l'exécution. Pour toutes ces fonctions, le processeur central emploie des signaux de commande C, pour commander la mémoire centrale et l'unité d'entrées/sorties. En réponse à ces signaux, les unités 2 et 3 envoient et reçoivent des informations que nous rangeons sous le vocable général d'opérantes O.

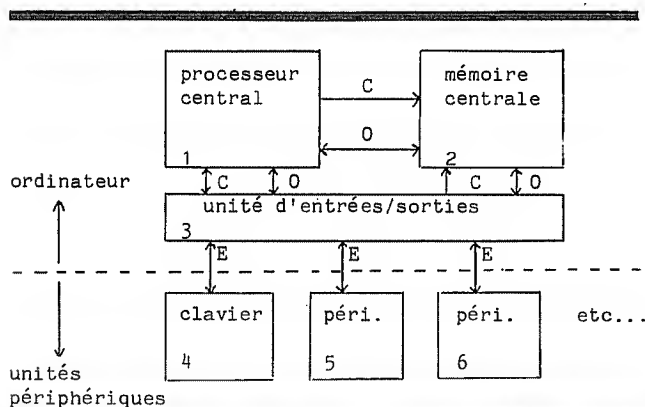


figure 1 - constitution d'un ordinateur.

L'unité 2, qui est la mémoire centrale, est constituée d'un ensemble de cellules. Dans les ZX, la mémoire centrale est subdivisée en MEM (ou ROM) et en MEV (ou RAM). Chaque cellule contient une information ; nous en distinguons deux sortes :

- 1 - Les instructions rangées séquentiellement en mémoire et dont l'ensemble constitue un programme. Le processeur central en commande l'extraction afin de les décodifier (MEM et MEV).
- 2 - Les opérantes sur lesquelles vont s'effectuer les différentes opérations décodées par le processeur (MEV uniquement).

Nous voyons que la mémoire a un rôle passif mais très important. Elle subit les commandes C aussi bien de la part du processeur que de l'unité 3. Ces commandes sont de deux sortes :

- 1 - Les commandes de lecture, auquel cas la mémoire fournit l'information d'une cellule au demandeur, (MEM et MEV).
- 2 - Les commandes d'écriture et la mémoire enregistre dans une cellule l'information venant des unités 1 ou 3 (MEV uniquement).

Tel quel, l'ordinateur fonctionne parfaitement. Cependant tout ce qu'il fait reste inconnu. Pour compléter notre ordinateur, il lui faut une 3ème unité, dite d'entrées/sorties qui interface les deux unités précédentes au monde extérieur. Elle établit des communications entre l'ordinateur et les unités extérieures qu'on appelle unités périphériques à l'aide de signaux d'échange E. Dans les ZX, cette fonction est remplie par le circuit intégré spécialisé et spécifique à Sinclair. Dans les ZX les périphériques prévus, donc interfacés, sont : un clavier, un enregistreur, l'imprimante Sinclair, un modulateur pour T.V. et un petit H.P. (Spectrum). La présence d'un connecteur arrière permet le raccordement à tout autre périphérique : mémoire supplémentaire, imprimante commerciale, manette de jeux, etc... mais chaque fois avec un circuit d'interfaçage adéquat.

La configuration d'un système de traitement d'informations est déterminée par l'utilisateur. En fonction du problème à résoudre, l'utilisateur établit la liste et les caractéristiques des unités dont il a besoin et la manière de les faire travailler. L'utilisateur doit donc aborder plusieurs techniques pour atteindre son but.

Le matériel (ou hardware) qui réunit la technologie, les circuits logiques, etc... est tout ce qui est matériel, visible, palpable. C'est le domaine de l'électronicien.

Le logiciel (ou software) qui réunit le langage de programmation, le système d'exploitation, etc... est tout ce qui est immatériel, invisible, spirituel. C'est le domaine du programmeur.

Signalons à ce sujet, deux particularités des ZX qui rendent son utilisation tellement agréable. Le fait d'introduire les mots-clés à l'aide d'une seule touche rend la programmation plus rapide.

Deuxièmement, le contrôleur de syntaxe qui signale les erreurs d'une ligne éditée avant son introduction dans un programme. (Ne pas confondre avec les compte-rendus d'erreur que tous les micro-ordinateurs possèdent avec plus ou moins de codes et de texte). Ces deux particularités réduisent fortement les erreurs de programmation et le temps qu'on consacre à leurs recherches.

Marcel Henrot

# langages

# LOGIGRAPH

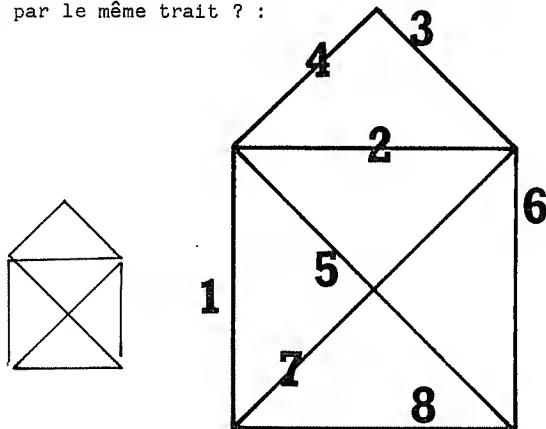
Yves Leclerc, le célèbre auteur canadien d'ouvrages sur l'informatique individuelle, qualifie le Langage LOGO, inventé par Seymour Papert, de "convivial" ; et ce terme est assez bien trouvé pour ce langage naturel, d'accès facile et spécialement adapté à l'initiation scolaire.

Les listes de LOGO sont très lisibles puisque chaque instruction symbolise nettement une action concrète orientée vers le graphisme : le programmeur s'identifie à une "petite tortue" qui commence à marcher au milieu de l'écran et qui change de direction, avance de tant de points, rechange de direction etc, et accomplit ainsi un parcours qui définit un graphisme plus ou moins élaboré.

Les interpréteurs LOGO sont très consommateurs de mémoire puisque chaque instruction représente une action complète ; il en résulte que ce langage, et c'est un paradoxe qu'on peut relever, est peu répandu sur les ordinateurs domestiques alors qu'il est tout indiqué pour l'apprentissage informatique des enfants. Il n'existe pas, à notre connaissance, de cassette développée pour Sinclair, mais ceci ne tardera sans doute pas. Pour vous préparer à cet événement, B. Clergeot vous propose un programme d'initiation aux principes du LOGO sous la forme d'un logiciel de simulation avec un dictionnaire déjà étoffé.

## Mise en oeuvre du programme

Un fois la liste entrée et l'ordre RUN lancé, l'ordinateur demande de rentrer des chaînes de caractère : il faut rentrer successivement les différentes instructions qui composent le petit programme de démonstration (n° 1) dessinant un rectangle : attention ne pas inscrire les numéros de ligne, le Spectrum numérote automatiquement. Quand toute la liste est entrée, introduire "1" et vérifier qu'il n'y a pas de faute dans la liste ; puis "a" et voyez le rectangle se dessiner. Voilà vous avez compris le principe... Etudiez attentivement les deux autres programmes également reproduits pour vous donner des idées, et, en choisissant vos instructions dans le "dictionnaire - logigraph", avec l'aide des indications données dans "l'éditeur" pour la correction de vos listes, essayez maintenant de programmer ce petit casse-tête classique : comment dessiner une maison avec son toit sans repasser deux fois par le même trait ? :



## LOGIGRAPH

\* Capacité : 100 lignes  
 \* Variables : 10 indicées x : v (x)  
 \* Label : 20 indicées x : Label x  
 \* Longueur d'une ligne : Maximum 29 caractères

## EDITEUR

Le programme se présente sous forme de lignes numérotées.

Pour entrer une instruction on n'entre par le n° de ligne : on les entre à la suite, puis on retire où ajoute des lignes.

Instructions : Mode direct

1 : liste les 22 premières lignes.

1 x : liste à partir de x.

i : fait une place entre deux lignes en décalant la ligne du dernier listing vers le bas.

d : supprime la ligne du dernier listing et remonte d'une ligne celles du dessous.

e : exécute le programme.

Sauve : sauve le programme sur cassette.

Oublier : perd le programme.

Variable v (x) : affiche la variable V(x).

## DICTIONNAIRE "LOGIGRAPH"

Instructions : mode interprété par le BASIC.

V(x)=n : la variable V(x) prend la valeur n.

Direction n : la direction en cours devient n.

Position nnn,nnn : le point de début du graphisme est est le PLOT nnn,nnn.

Efface : efface l'écran.

Papier x : le fond devient de la couleur de x.

Encre x : l'encre est de la couleur de x.

Afficher en nn,nn : affiche ce qui est écrit à la ligne suivante aux coordonnées d'écran nn,nn.

Exemple : 10 AFFicher en 10,10

11 Bonjour je suis le Spectrum

Label x : positionne un label où il sera fait un saut.

Son : fait un son bref et aigue.

Avance x : avance dans la direction en cours de x

Avance V(x) : avance de V(x)

Gauche x : la direction en cours vire à gauche de x degrés

Gauche V(x) : la direction en cours vire de V(x) degrés

Droite x : la direction en cours vire à droite de x degrés

Droite V(x) : la direction en cours vire à droite de V(x) degrés

Fin : fin du programme

Va au label x : saut incondtionnel au label x

Si V(x)=0 va au label x : saut incondtionnel : si la variable V d'indice x est égale à 0

Liste x : fait en programme la liste des 22 lignes suivant x, x inclus.

Calcul de v(x) = ..... : la variable V(x) prend la valeur du résultat de ce qui suit le égal.

Presser le space arrête le programme.

Dans les calculs vous pouvez utiliser la variable.

h a s a r d qui est comprise entre 0 et 1.

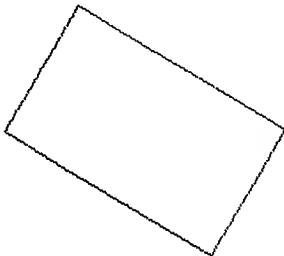
En cas d'arrêt du programme : faire GOTO edit.



```

1 effacer
2 direction 30
3 position 170,10
4 avance 90
5 gauche 90
6 avance 150
7 gauche 90
8 avance 90
9 gauche 90
10 avance 150
11 fin

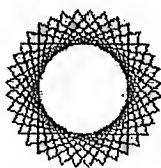
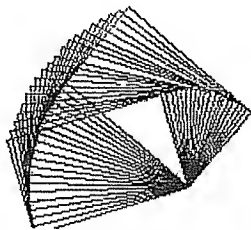
```



```

1 effacer
2 position 170,050
3 v(1)=40
4 v(2)=20
5 label 1
6 direction v(1)
7 avance 50
8 gauche 90
9 avance 100
10 gauche 90
11 avance 60
12 gauche 90
13 avance 100
14 calcul de v(1)=v(1)-3
15 calcul de v(2)=v(2)-1
16 si v(2)=0 va au label 2
17 va au label 1
18 label 2
19 fin

```



```

1 papier 0
2 encr 7
3 effacer
4 direction 0
5 position 140,050
6 afficher en 00,05
7 programme en logigraphe
8 v(1)=35
9 label 2
10 avance 80
11 gauche 110
12 calcul de v(1)=v(1)-1
13 si v(1)=0 va au label 1
14 va au label 2
15 label 1
16 son
17 papier 7
18 encr 0
19 fin

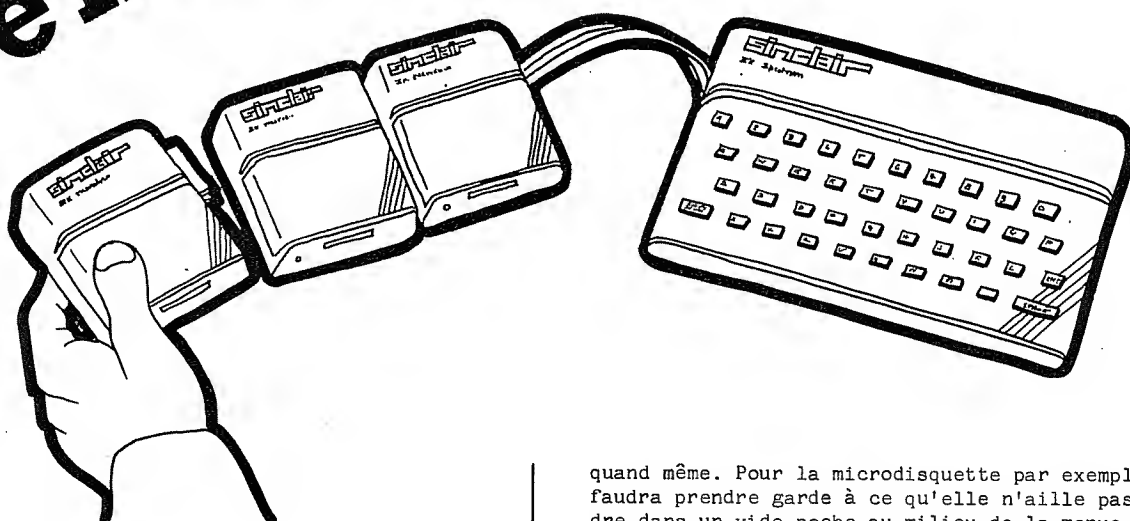
```

```

5 PRINT AT 0,5:"@CLERGEOT Ber
trand 1983"
6 PRINT AT 5,10:"LOGIGRAPHE"
7 REM *****logigraphe*****
8 LET O=0
9 DIM K=360/(2*PI)
10 DIM I$(100,29)
11 DIM L(20)
12 DIM v(10)
13 LET edit=8000
14 GO TO 5000
15 REM *****interpreteur*****
16 LET co=1
17 PLOT 128,87
18 LET c$=I$(co)
19 LET hasard=RND
20 IF c$(1 TO 6)="avance" THEN
GO SUB 1000
21 IF c$(1 TO 7)="gauche" OR
c$(1 TO 7)="droite" THEN GO SUB
1100
22 IF c$(1 TO 3)="fin" THEN GO
TO 5000
23 IF c$(1 TO 6)="efface" THEN
CLS
24 IF c$(1 TO 9)="direction" T
HEN LET O=VAL c$(11 TO )
25 IF c$(1 TO 6)="position" TH
EN PLOT URL c$(10 TO 12),VAL c$(
14 TO 16)
26 IF c$(1 TO 11)="va au label
" THEN LET co=(VAL c$(13 TO ))
27 IF c$(1 TO 2)="v(" THEN LET
v(VAL c$(3))=VAL c$(6 TO )
28 IF c$(1 TO 5)="si v(" THEN
GO SUB 4000
29 IF c$(1 TO 9)="calcul de" T
HEN LET v(VAL c$(13))=VAL c$(16
TO )
30 IF c$(1 TO 6)="papier" THEN
PAPER VAL c$(8 TO )
31 IF c$(1 TO 5)="encr" THEN
INK VAL c$(7 TO )
32 IF c$(1 TO 5)="liste" THEN
LET p=VAL c$(7 TO ); GO SUB 6100
33 IF c$(1 TO 8)="afficher" TH
EN PRINT AT VAL c$(13 TO 14),VAL
c$(16 TO 17);I$(co+1); LET co=c
o+1
34 IF c$(1 TO 3)="son" THEN BE
EP OS,30
35 IF INKEY$="" THEN PRINT AT
21,0;"arret en ";co; GO TO 5000
36 LET co=co+1
37 GO TO 105
38 IF c$(8)="v" THEN ORRU SIN
(D/K)*v(VAL c$(10)),COS(D/K)*v(
URL c$(10)); RETURN
39 ORRU SIN(D/K)*URL c$(8 TO
),COS(D/K)*VAL c$(8 TO ); RETUR
N
40 IF c$(1)="d" THEN LET d=d+(
URL c$(8 TO )); RETURN
41 IF c$(1)="g" THEN LET d=d-(
URL c$(8 TO )); RETURN
42 GO SUB 9000
43 RETURN
44 PRINT "preparation des vabs
"
45 FOR n=1 TO 100
46 IF I$(n)(1 TO 5)="label" TH
EN LET L=VAL I$(n)(7 TO );n
47 NEXT n
48 GO TO 100
49 IF v(VAL c$(6))=0 THEN LET
co=(VAL c$(23 TO ))
50 RETURN
51 REM *****editeur*****
52 LET p=1
53 INPUT C$: IF C$="" THEN GO
TO 5020
54 LET C$=C$+" "
55 LET T=0
56 IF C$="sauve" THEN SAVE "p
rog. logi" ORTA I$(1); GO TO 5000
57 IF C$="lire" THEN LOAD "pr
og. logi" ORTA I$(1); GO TO 5000
58 GO TO 5000
59 LET I$(p)=C$
60 IF C$="oublier" THEN RUN
61 PRINT P;TAB 3;I$(p)
62 LET P=P+1; IF P=101 THEN LE
T P=100
63 GO TO 5020
64 IF c$(1 TO 2)="l " THEN GO
SUB 6100
65 IF c$="i " THEN GO SUB 6200
66 IF c$="d " THEN GO SUB 6300
67 IF c$="s " THEN GO TO 3000
68 IF c$(1 TO 3)="var" THEN GO
SUB 6400
69 IF T=1 THEN GO TO 5020
70 GO TO 5030
71 CLS: IF LEN C$=2 THEN LET
C$=C$+"1"
72 LET p=URL c$(3 TO )
73 IF p>100 THEN GO TO 6125
74 FOR n=p TO p+21
75 IF n>100 THEN GO TO 6125
76 IF I$(n)(1)=" " THEN GO TO
6125
77 PRINT n;TAB 3;I$(n); NEXT n
78 LET T=1
79 PRINT AT 0,0;
80 RETURN
81 LET T=1
82 FOR n=99 TO p STEP -1
83 LET I$(n+1)=I$(n)
84 NEXT n
85 LET I$(p)=""
86 LET C$="l "+STR$ p; GO SUB
6100
87 RETURN
88 FOR n=p TO 99
89 LET I$(n)=I$(n+1); NEXT n
90 LET I$(100)=""
91 LET C$="l "+STR$ p; GO SUB
6100
92 RETURN
93 PRINT "v(";c$(12);")=";v(UR
L c$(12)); LET T=1; RETURN
94 CLS: PRINT "instruction: "
; c$; PRINT "erreur en ligne ";co
; PRASE 0; LET C$="l "+STR$ co;
GO TO 5021
95 IF d>360 THEN LET d=d-360
96 IF d<0 THEN LET d=360+d
97 RETURN
98 FOR n=0 TO 2*PI STEP .1

```

## Le microdrive, enfin !



Certains d'entre vous vont dire : "ORDI-5 nous parle du Microdrive alors qu'on en est encore à attendre le Spectrum". Pourtant nous ne résistons pas au plaisir de vous en parler ; d'abord parce que ce n'est pas si prématuré : les Anglais commencent à être livrés d'une manière régulière, ensuite parce qu'il multiplie à tel point les possibilités du Spectrum que certains pourraient bien attendre qu'il soit là pour trouver un grand frère à leur ZX81 avant de tourner trop vite vers une certaine concurrence (suivez mon regard...) ; enfin et surtout parce que nous avons trop envie de vous décrire ce bijou que la rédaction a eu le privilège de tenir dans les mains grâce à la gentillesse du directeur général de la firme Sinclair lui-même.

### IL TIENT DANS LE CREUX DE LA MAIN

Un vrai bijou donc, c'est l'impression que fait le microdrive quand on le sort de sa boîte. On l'avait bien vu en photo et on le savait petit pas à ce point

quand même. Pour la microdisquette par exemple, il faudra prendre garde à ce qu'elle n'aille pas se perdre dans un vide-poche au milieu de la menue monnaie : elle est vraiment microscopique. Mettre un timbre dans une boîte d'allumettes de cuisine, ce sont à peu près les dimensions. Soyons plus précis, le drive lui-même mesure 90 mm sur 85 mm avec une épaisseur de 40 mm. La microdisquette, que nous appellerons la cartouche, est un petit rectangle de 45 sur 35 mm avec 7 mm d'épaisseur.

Le Microdrive n'a pas seulement la taille du bijou il en a la beauté : un design qui rappelle le Spectrum lui-même, une belle couleur noire avec le label en arc-en-ciel sur le coin inférieur droit, et tous les éléments en sont ajustés avec une grande précision : une fois la cartouche insérée dans la fente prévue à cet effet il n'y a pas le moindre jeu.

Le Microdrive en revanche s'éloigne d'une troisième caractéristique propre aux bijoux : le prix ; il est en effet actuellement vendu en Angleterre pour la somme de 49 livres soit 600 F : étant donné qu'il tient les promesses techniques (annoncées voilà quelques temps déjà !), on a l'impression que c'est comme si on pouvait acheter une voiture de sport pour le prix d'une mobylette. Nous avons eu très envie d'ouvrir le ventre du microdrive pour voir comment on pouvait faire aussi petit et puis nous nous sommes tout de même raisonnés : ce n'est pas le moment de tout casser, avant de faire de la chirurgie, attendons de pouvoir se procurer un microdrive de rechange dans la première boutique spécialisée comme une simple cassette vierge : la leçon d'anatomie ce sera pour un prochain numéro d'ORDI-5.

## 100 FOIS PLUS RAPIDE !

Venons en à l'essentiel, la capacité et la vitesse : les microdrive sont des cartouches à bande sans fin qui reprennent une technique déjà utilisée dans les "wafertape" mais nettement miniaturisée : la bande est plus étroite que celle d'une "microcassette" classique ; il y a environ 5 mètres de ruban qui passent à la vitesse de 80 centimètres à la seconde devant la tête de lecture : il s'ensuit qu'il suffit environ d'une dizaine de secondes pour lire la totalité de la bande sur un tour.

Le microdrive est prévu pour 100K (en fait un peu moins : 85 K étant un minimum toujours garanti).

Un programme de 48K est lu en moins de 5 secondes si la bande est positionnée en début de fichier : la lecture de programmes pré-enregistrés se fait avec une vitesse multipliée par un facteur 100 ! Un bruit de feulement assez fort et assez soutenu en dit long sur la vitesse de déplacement.

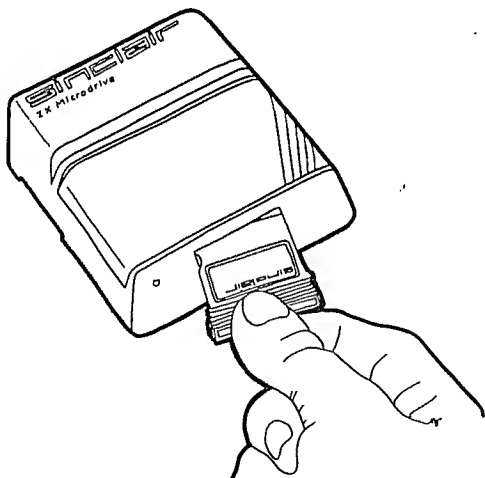
## LA DEUXIEME MERVEILLE

Tel quel, le drive ne peut pas être utilisé en conjonction directe avec le Spectrum, il est nécessaire d'ajouter une interface, dite "interface 1" qui constitue elle aussi une merveille de rapport possibilité/prix puisque pour 350 FF on dispose d'un système non seulement capable de piloter une bande sans fin mais aussi de permettre les développements de télécommunication et le couplage avec des imprimantes rapides : il s'agit de l'interface RS 232.

Cette merveille est la partie modeste du dispositif ; modeste parce qu'elle vient s'accoler à l'arrière du Spectrum et en dessous (ce qui en passant donne à l'appareil une position plus ergonomique), mais modeste également car l'essentiel de la publicité n'a pas porté sur cet élément qui constitue pourtant une bonne part de la prouesse technique de Sinclair : proposer une interface RS232 à ce prix. Il est clair que nos amis Anglo-américains peuvent avec un minimum d'argent se doter d'un ordinateur absolument complet et sans équivalent dans cette catégorie.

## BOND DANS LE DOMAINE PRO

Ces capacités considérablement accrues permettent d'envisager sans sourire de véritables applications professionnelles pour le Spectrum - nous entendons par véritables celles qui ne demandent pas à l'utilisateur un trop gros effort de manipulation, puisqu'aussi bien le ZX81 est capable de supporter des applications



professionnelles importantes quand on est capable de surmonter la rusticité des opérations de sauvegarde ou chargement. --

Un domaine dans lequel l'amélioration de la vitesse de chargement peut avoir une influence décisive est celui des jeux : la frustration que l'on éprouve à voir les marbrures de l'écran pendant quelques minutes avant de pouvoir jouer vont disparaître ; on peut aussi penser à des échanges de jeux et de logiciel entre clubs en glissant simplement dans une enveloppe la microcartouche de quelques grammes : et nous parlons du courrier "avion" puisque la cartouche ne pèse que 8 grammes !

Entre un Spectrum ordinaire et un Spectrum muni de son microdrive et de son interface il n'y a pas qu'une simple différence de degré, il y a une véritable opposition de nature : on a affaire à un autre ordinateur comme on s'en rend tout de suite compte avec l'utilisation d'une batterie tout à fait nouvelle de commandes.

Quiconque n'a pas utilisé les disquettes des gros ordinateurs de table mettra même du temps à se rendre compte de toutes les possibilités qu'il y a à utiliser en accès immédiat 100 K de mémoire en masse, en ouvrant et en fermant les fichiers, en appelant en désordre différents programmes de travail etc...

Après un certain temps d'utilisation on s'aperçoit que le véritable drame consiste à ne disposer que d'une seule microcartouche livrée d'origine avec l'appareil : il s'agit encore pour l'instant d'une denrée rare ; on voudrait pouvoir tout mettre dans ces 100 K, et l'on arrive vite à saturation (mais oui !). Vivement la production de masse ! Signalons une difficulté à ce niveau : la transposition de programmes Spectrum enregistrés sur cassette, qui font usage de routines en langage machine, nécessitent une adaptation, car le branchement de microdrives modifie sensiblement les zones de mémoire RAM ; cet inconvénient n'est pas rencontré avec les programmes tout BASIC.

## EXPLICATION DES MYSTERIEUSES INSTRUCTIONS DU CLAVIER

CAT : L'ordre CAT affiche sur l'écran en 10 secondes les titres de tous les fichiers (programmes ou données) contenues sur la cartouche à concurrence de 50 titres.

FORMAT : Prépare la microcartouche en lui donnant un nom.

ERASE : Efface sur la cartouche les programmes ou données que l'on veut écraser.

OPEN # et CLOSE # ouvrent et ferment les fichiers : un fichier fermé ne peut être écrasé par erreur.

MOVE : Permet de transférer les informations d'un canal à un autre (écran, clavier, microdrive, sont des canaux).

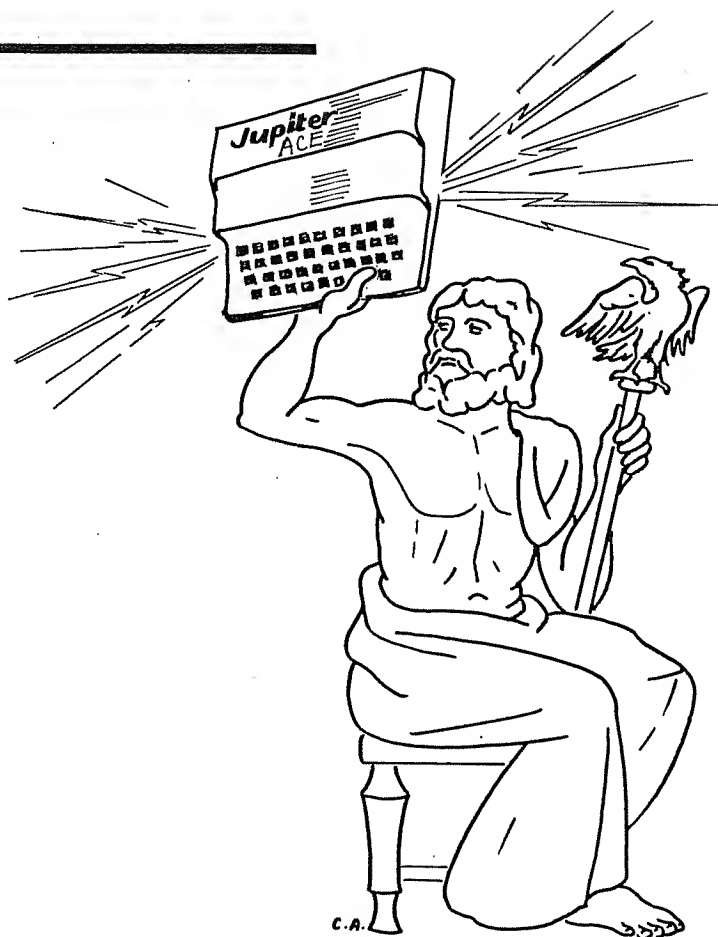
Le paramétrage de toutes ces commandes, comme des commandes classiques LOAD, SAVE et VERIFY, peut être assez complexe étant donné la multiplication des possibilités par exemple.

SAVE "m"; "ORDI-5" LINE 1 signifie : sauvegarder sur la première cartouche le programme ORDI-5 de telle sorte que, chargé, il s'exécute à la première ligne.

Henri Davignat

## Modulons

## notre FORTH



Il y a deux "environnements" très différents pour les programmeurs : l'interprétation et la compilation qui conduisent à des méthodes de programmation très différentes.

La première méthode est née avec les micro-ordinateurs programmables en BASIC. Ce dernier interprété, permet une programmation aisée, facilite les modifications et autorise la programmation "fouilli" (il y invite même d'ailleurs). Son principal avantage est qu'il est conversationnel lors de l'écriture des programmes.

La deuxième méthode date des ordinateurs où les programmes étaient entrés par des cartes perforées, puis compilés et enfin exécutés. La programmation ainsi faite est souvent fastidieuse car longue mais a l'avantage de fournir des programmes plus "propres", lisibles et où l'on sait toujours exactement ce qui se passe (enfin presque toujours !).

FORTH présente cet avantage à peu près unique de combiner l'interprétation pour l'élaboration des programmes et la compilation pour leur exécution parce que la programmation progresse d'une façon modulaire : on ne peut pas travailler une nouvelle instruction (mot) sans avoir compilé les instructions précédentes; ceci permet des corrections rapides du programme, des essais de petits morceaux (grâce à sa modularité) et une programmation très fortement structurée (pour qui veut s'en donner la peine).

Le programme peut donc, lorsqu'il a été dégrossi sur le papier, être entré directement dans le micro-ordinateur et essayé sans plus attendre.

Malheureusement, et c'est là le principal défaut de FORTH, à l'état brut, sans commentaire, un programme est illisible. Il faut donc toujours fournir l'effort de documenter chaque module sous peine d'être incapable de s'en servir huit jours après.

Nous allons illustrer cette idée de structuration modulaire en programmant un petit jeu simple et connu :  
- L'ordinateur génère un nombre aléatoire que le joueur doit trouver en proposant successivement des nombres. Des indications de supériorité ou d'infériorité lui sont données pour chaque tentative.  
Lorsque la réponse est trouvée, l'ordinateur donne le nombre de tentatives et termine.

### LE NOMBRE SECRET

L'analyse s'effectuera en deux étapes :

- d'abord l'analyse des modules du programme principal
- ensuite l'analyse des différentes instructions de chaque module.

Le tout sans effort logique démesuré.

### PROGRAMME

- Initialisation  
Nombre de propositions = 0
- Aléatoire (appel du programme de tirage au sort du chiffre)
- Proposition
- Si Egaux alors Terminé  
                                  sinon Supérieur ?
- Incrémenter le nombre de propositions  
(Incrémenter c'est ajouter 1)
- Attendre (pour éviter un effacement immédiat du résultat de la proposition)
- Reprendre à proposition
- Fin programme ;

### Aléatoire

(Ce sous programme est appelé par son nom dans le programme principal)

- Tirer un nombre aléatoirement ;
- Le mettre sur la pile
- Fin d'aléatoire ;



## Proposition

- Ecrire 'Entrez votre proposition :'  
 - Ecrire ' (nombre entre 1 et 50) '  
 - Attendre un nombre  
 - Si nombre > 50 alors écrire 'Erreur' et charger le nombre en 0

sinon mettre le nombre sur la pile

Fin proposition ;

## Egaux

Si les 2 nombres sur la pile sont égaux  
 alors écrire 'C'est gagné !'  
 écrire nombre proposition 'Coups'  
 mettre oui sur la pile (c'est pour dire au programme principal que c'est terminé)  
 sinon mettre non sur la pile

Fin égaux ;

## Supérieur ?

Si le nombre aléatoire est > à la proposition  
 alors écrire 'Trop grand !'  
 sinon écrire 'Trop petit !'

Fin supérieur ? ;

## Attendre

Faire 800 fois rien (une boucle qui ne fait rien que perdre du temps)

Fin attendre ;

## LE PROGRAMME

```

: RND (Construire ce mot comme dans le manuel du
      Jupiter Ace page 85)
: CHIFFRE (mot de création d'une entrée numérique)
  QUERY (Demande un nombre)
  NUMBER (Transfère sur la pile ce nombre et son
          adresse)
  DROP (Laisse tomber l'adresse)
;
0 VARIABLE NBRPROPOSITION (Crée la variable de stoc-
                          (kage du nombre de Proposi-
                          tions du Joueur. Initiali-
                          se la variable à 0)

(Génération du nombre à trouver en aléatoire)
: ALEATOIRE (le : met en mode compile et on crée le )
            (nom du nouveau module. )
  50 RND 1+ (le nombre sera entre 1 et 50 )
;          (fin de ce module. )

: PROPOSITION (Nouveau module de nom PROPOSITION )
  CLS (Efface l'écran )
  ." Entrez votre proposition "
  CR (Passe à la ligne suivante )
  ." Un nombre entre 1 et 50 puis retour chariot:" )
  (on vient d'écrire une question sur )
  (l'écran )
  CHIFFRE (on attend l'entrée du joueur )
;

(Test de l'égalité entre la proposition et le nombre)
: EGAUX (Nom du nouveau module )
  OVER (copie du nombre à trouver sur la pile )
  OVER (copie de la proposition sur la pile )
  = (test de l'égalité )
  IF CR ." GAGNE !!!!"(on écrit que le joueur a )
    (gagné )
    CR ." En " (on écrit le nombre de coups)
    NBRPROPOSITION @.(nécessaire pour gagner )
    ."coups"
    1 (on laisse 1 sur la pile car ils sont )
    (égaux )
  ELSE (sinon on laisse 0 sur la pile pour )
    0 (dire qu'ils ne sont pas égaux )
  THEN (on sort du module )
;

(Test si la proposition est supérieure ou inférieure)
: SUPERIEUR?

```

```

OVER (copie de nombre à trouver sur la pile)
> (la proposition est supérieure ? )
IF
  CR ."TROP GRAND !" ( oui )
ELSE
  CR ."TROP PETIT !" (non )
THEN
  (on sort du module )
;

```

(Module de temporisation)

```

: ATTENDRE (Nom du module )
  8000 (De 8000 on décrémente )
  0 (à 0 )
  DO (On fait une boucle qui ne fait rien )
  LOOP (On sort du module )
;

```

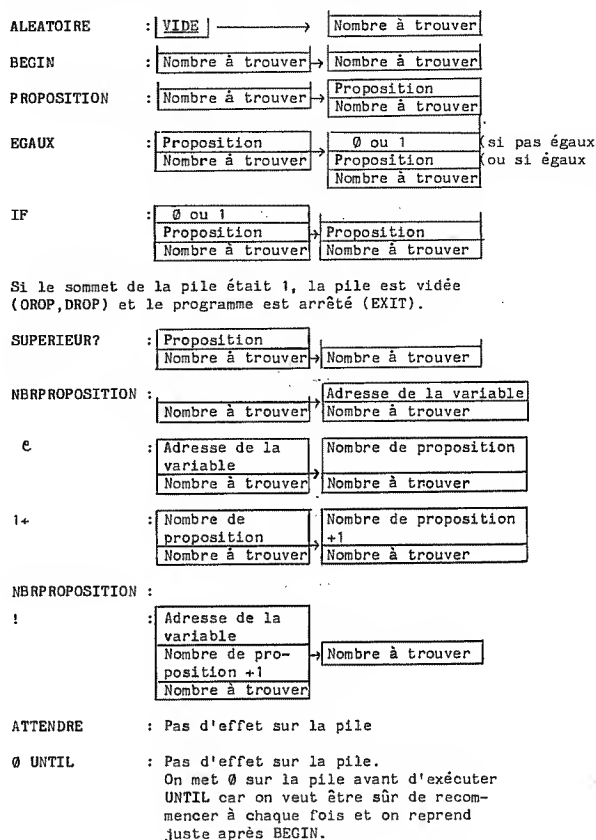
(Module principal du jeu)

```

: TROUVE (Nom d'appel du jeu )
  ALEATOIRE (On tire le nombre à )
  (trouver )
  BEGIN (Début des recherches )
  PROPOSITION (Le joueur tente un chiffre)
  EGAUX (A-t-il trouvé le bon )
  (chiffre )
  IF DROP DROP EXIT (Oui on vide la pile et on)
  (sort )
  ELSE SUPERIEUR? (Non trop grand ou trop )
  (petit ? )
  THEN
  NBRPROPOSITION @ (On met le nombre d'essais)
  1+ (sur la pile et on ajoute )
  NBRPROPOSITION ! (1 avant de le restocker )
  ATTENDRE (On attend avant de recom-
  0 UNTIL (mencer à cause de l'effa-
  (cement. )
;

```

## CE QUI SE PASSE DANS LA PILE A CHAQUE INSTRUCTION

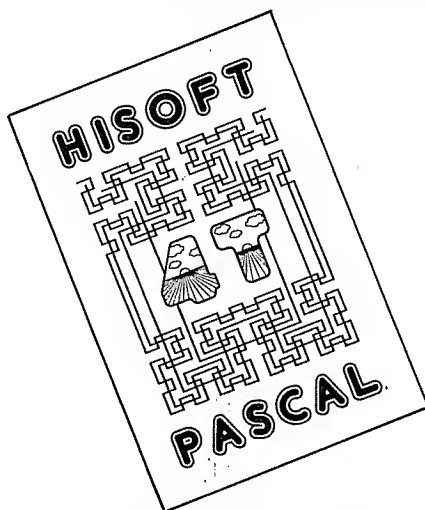


Régis Senegou

## PASCAL sur ZX

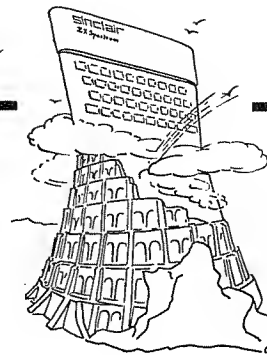
La cassette Pascal pour Spectrum 48 K de chez Hisoft est une réalisation de qualité professionnelle dont le succès est tel en Angleterre qu'aucun concurrent ne s'est encore risqué à proposer une cassette du même type et que c'est plutôt celle-ci qui est transposée pour d'autres matériels que le Spectrum.

Chacun sait qu'un compilateur Pascal tient en mémoire une place considérable : l'étude de ce langage ne pouvait donc être menée valablement que sur des ordinateurs moyens tels l'Apple. On peut maintenant véritablement apprendre le Pascal sur un ordinateur domestique car le compilateur et le programme d'interface n'occupent qu'une vingtaine de K de mémoire. Nous disons bien apprendre le Pascal et non simplement programmer en Pascal ; en effet, quoi qu'en dise la documentation technique qui prétend que cette cassette est faite pour les gens qui savent déjà programmer dans ce langage, nous estimons, nous, qu'elle est aussi parfaitement adaptée à l'étude progressive de celui-ci étant donné la manière particulièrement explicite dont l'éditeur signale les fautes de syntaxe.



Cette cassette a donc une portée assez générale et, à condition d'avoir fait au préalable l'acquisition du supplément mémoire nécessaire, il s'agit d'un "package" qui en donne énormément pour pas cher : on peut sans prendre de risque mettre quiconque à défi de commencer à programmer en Pascal pour moins de 300 F.

Comme la chose se produit très souvent pour les cassettes de grande qualité, le manuel n'est pas, mais pas du tout, à la hauteur du programme. Ce qui vérifie une fois de plus un phénomène souvent rencontré : que d'excellents programmeurs peuvent être de piètres pédagogues et de mauvais écrivains : chacun son métier, évidemment ! Il en résulte de très nombreux tâtonnements avant de savoir quelle touche provoque telle réaction ou quel est l'effet de telle fonction à tel stade : pour parcourir les Anglais nous dirions que la documentation est "extremely confusing". Nous avons mis des heures à comprendre comment on pouvait déclencher l'option imprimante du compilateur pour vous lister le petit programme de résolution d'équation du second degré ci-dessous : il faut faire (X!P<sub>X</sub>) dans un commentaire, cela ressemble à un phylactère d'injure dans une bande dessinée...



Tout ce qui pourrait transformer le manuel en un ouvrage praticable est absent : il n'y a pas d'organigramme général pour illustrer la hiérarchie des différentes commandes ou le moyen simple de passer de l'un des trois états (BASIC, Editeur, Compilateur) à l'un des deux autres ; il n'y a pas un seul exemple correctement développé pour faire la découverte pas à pas des différentes possibilités de la cassette ; il n'y a pas d'index permettant de s'y retrouver dans le livre et enfin il n'y a aucun plan rationnel ; zéro donc pour le manuel.

10 PROGRAM EQ;

```

20 VAR A,B,C:REAL;
30 D,R,I:REAL;
40 BEGIN
50 READ (A,B,C);
60 R:=-B/(2*A);
70 D:=B*B-4*A*C;
80 I:=SQRT(ABS(D))/(2*A);
90 IF D>0
100 THEN WRITELN('Les raci
des sont: ',R+I,' et ',R-I)
110 ELSE IF D=0
120 THEN WRITELN('Racine d
uble: ',R)
130 ELSE WRITELN('Racines
complexes: ',R,'+i',I,' et ',R,'-
i',I)
140 END.
End Address: AD68

```

10 PROGRAM AB;

```

20 VAR A,B:INTEGER;
30 BEGIN
40 READ (B);
50 FOR A:=1 TO B DO
60 BEGIN
70 WRITE('*');
80 END;
90 END.
End Address: AB72

```

```

10 INPUT A
20 FOR B=1 TO A
30 PRINT "*";
40 NEXT B

```

UN LOGICIEL D'EXCELLENCE !

Heureusement si l'on s'accroche deux bonnes heures en substituant sa logique à celle des rédacteurs on découvre l'une après l'autre les extraordinaires qualités de ce logiciel et l'on prend vite beaucoup de plaisir à utiliser les différentes fonctions d'édition qui ont un air très professionnel (il en est ainsi) possible de commander l'incrément automatique des instructions comme le renumérotage ou l'effacement partiel et sélectif du programme, etc.) ; le plaisir s'accroît naturellement lorsque l'on commande la compilation qui s'effectue sous nos yeux ligne par ligne à une vitesse très appréciable ; enfin l'exécution des programmes objet est d'une grande rapidité comparé à l'équivalent en BASIC sans atteindre naturellement à la rapidité de l'Assembleur "cousu main". Ce gain de rapidité est d'un facteur 5 à 10 ainsi que nous l'avons testé sur le petit programme numéro 2 qui consiste à couvrir l'écran d'astérisque (1 seconde contre 5).

Nous aurons certainement l'occasion de vous reparler de ce logiciel et nous faisons même un "appel au peuple" pour quelques beaux programmes réalisés de cette manière à insérer dans les colonnes d'ORDI-5.

X. L. B.

# technique

## Réalisez

### un port de

### sortie 8 bits

Avec une dizaine de circuits intégrés vous pouvez réaliser une extension qui permet au ZX de commander des éléments extérieurs, de petits automatismes par exemple.

#### PRINCIPE DE FONCTIONNEMENT

On dispose de 8 sorties prenant chacune un état 1 ou 0, ces sorties sont utilisables individuellement ou sous la forme d'un octet : sous cette dernière forme la valeur de la sortie sera comprise entre 0 et 255 (00 à FF en hexadécimal).

Ce circuit n'occupe qu'une seule adresse pour sortir l'octet de donnée ; cette adresse est sélectionnée par straps entre 0 et 65535 :

On compare les 16 bits d'adresse (A0...A15) avec la valeur sélectionnée par 16 straps (qui forment l'adresse binaire désirée). S'il y a égalité de valeur, on mémorise l'octet correspondant à cette adresse et présent sur le bus des données.

Donc le circuit réagit simplement à la présence de l'adresse voulue sans se préoccuper des signaux de contrôle du microprocesseur tels que MREQ, RD ou WR.

#### DESCRIPTION DU SCHEMA

Les bus d'adresses et des données sont tout d'abord envoyés sur des buffers (IC1...IC4) qui permettent de ne pas trop charger les sorties du microprocesseur et de pouvoir utiliser ces informations pour d'autres extensions sans perturber le bon fonctionnement du ZX. Depuis la sortie des buffers, les 16 bits d'adresse vont sur un comparateur 16 bits (sur les entrées A). Ce comparateur est réalisé avec 4 (IC6...IC9) comparateurs 4 bits (74LS85) montés en cascade.

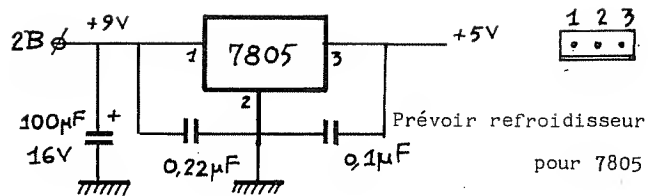
On sélectionne l'adresse désirée en positionnant les straps branchés sur les entrées B du comparateur. La sortie 6 de IC9 sera au niveau haut si, et seulement si, toutes les comparaisons sont vraies.

On transforme la sortie du comparateur en impulsion par l'intermédiaire de R1 et C1. Cette impulsion sert à charger les 8 bascules (74LS75 x 2) servant de mémoire pour les données. Les straps S1 et S2 sont prévus pour d'autres extensions et permettent si besoin est de bloquer les buffers.

#### RECOMMANDATIONS POUR LE MONTAGE

Pour la construction du montage on suivra les recommandations suivantes :

- Prévoir un régulateur +5V pour l'alimentation de ce circuit (consommation  $\approx$  200 mA).



face métallique vue côté connexions du 7805 :

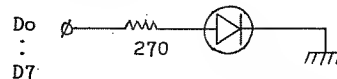
- Mettre un condensateur de 0,1µF en parallèle sur l'alimentation des CI tous les 2 circuits.
- Ramener les 8 fils de sortie ainsi que +9V, +5V et 0V sur un connecteur (ou une barrette de dominos) pour l'utilisation.
- A la place des 16 straps on peut installer :
  - . 2 blocs d'interrupteurs en boîtier DIL 16 broches
  - . 4 roues codeuses hexadécimales.

#### LISTE DES COMPOSANTS

74LS75 x 2  
74LS85 x 4  
74LS367 x 4  
7805 x 1 refroidisseur  
Résistance 100Ω x 1  
Condensateur : 100µF/16V x 1  
1µF/16V x 1  
0,1µF x 6  
0,22µF x 1

#### UTILISATION DE L'EXTENSION

Pour se familiariser avec le montage on peut brancher 8 LED en sortie :



Puis on sélectionne une adresse d'utilisation.

Dans le manuel du ZX, il est indiqué que l'adresse 16417(d) est inutilisée ; nous l'utiliserons donc pour les essais.

Pour passer d'une adresse décimale à son équivalent en binaire 16 bits on peut procéder comme suit :

$$16417(d) = 256 \times 64 + 33$$

Page 181 à 187 du manuel du ZX, on trouve la correspondance entre code hexa et code décimal, qui donne :

$$64(d) = 40(h) \text{ et } 33(d) = 21(h) \\ \text{donc } 16417(d) = 4021(h)$$

D'autre part, l'équivalent binaire des chiffres hexa (0...F) est :

0	0000	8	1000
1	0001	9	1001
2	0010	A	1010
3	0011	B	1011
4	0100	C	1100
5	0101	D	1101
6	0110	E	1110
7	0111	F	1111

4021(h) s'écrit donc : 0100 0000 0010 0001  
(A15) (A0)

et l'on positionnera les straps en conséquence en sachant qu'un strap mis représente un 0.  
Entrer le programme suivant :

```

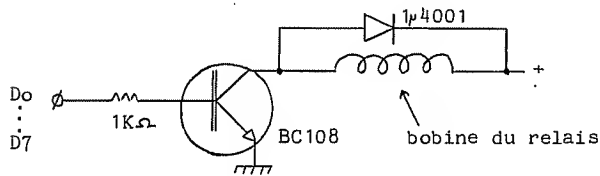
10 FOR A = 0 TO 255
20 POKE 16417,A
30 NEXT A
et RUN

```

On verra les LED s'allumer en fonction de la valeur binaire de A.

#### APPLICATIONS

##### 1) - Commande de relais :



##### 2) - Lecture d'une case mémoire contenant une variable.

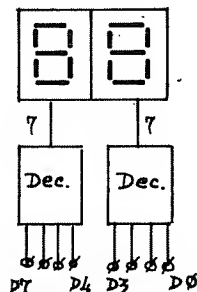
Dans la mise au point d'un programme en langage machine, on se sert souvent de cases mémoires pour stocker une variable de calcul.

Exemple : 2 D (26000),A

On aimerait voir les différentes valeurs de la variable pendant l'exécution du programme, ce qui est souvent difficile à obtenir facilement. L'extension rendra de grands services, car positionnée sur cette adresse (dans l'exemple : 26000(d)), elle nous indiquera le nouveau contenu à chaque changement.

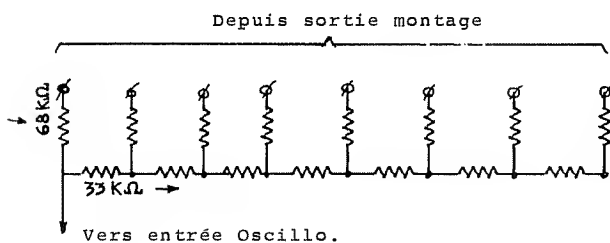
Il est possible, à la place des 8 LEDs en sortie qui ne sont pas commodées à lire, de monter 2 décodeurs hexadécimaux pour 2 afficheurs 7 segments. Chaque décodeur sera commandé par 4 bits.

Exemple :



3) - Commande d'un autre appareil (exemple : terminal vidéo) en appliquant l'octet de sortie à l'entrée parallèle. Il suffira pour la commande de faire succéder les différentes valeurs. Peut permettre de calculer en mode FAST et d'afficher sur un autre écran les résultats et sur un format qui n'est pas limité à 32 colonnes. Il faudra bien évidemment utiliser des routines en langage machine.

##### 4) - Utilisation de l'informatique pour l'électronique.



A l'aide du petit montage ci-contre et avec un oscilloscope - lorsque l'on est équipé - on peut visualiser des signaux de toutes les formes.

#### Programme d'application :

```

10 FOR A = 0 TO 255
20 POKE 16417,A
30 NEXT A
40 FOR A = 255 TO 0 STEP -1
50 POKE 16417,A
60 NEXT A
70 GOTO 10

```

mode FAST et RUN permet de produire des signaux triangulaires de fréquence très basse. La vitesse est limitée par le BASIC.

On peut augmenter la vitesse en utilisant une routine en langage machine.

Taper le programme suivant :

```

1 REM XXXXXXXXXXXX... (32 fois un caractère)
10 FOR A = 16514 TO 16544
20 INPUT B
30 POKE A,B
40 NEXT A
50 RAND USR 16514
60 GOTO 50

```

Rentrer les nombres suivants lors de l'exécution du programme :

6,255,120,50,33,64,16,250,201,STOP

Détail du programme en assembleur :

```

LD B,255
LD A,B
LD(16417),A
DJNZ-6
RET

```

en mode FAST on fait RUN 50, et on observe sur l'oscilloscope des signaux en dents de scie (fréquence ≈ 400 Hz).

Pour produire des signaux triangulaires, rentrer les nombres suivants lors de l'exécution du programme :

6,255,120,50,33,64,16,250,50,33,64,60,254,255,56,248,205,70,15,56,235,201,STOP

Détail du programme en assembleur :

```

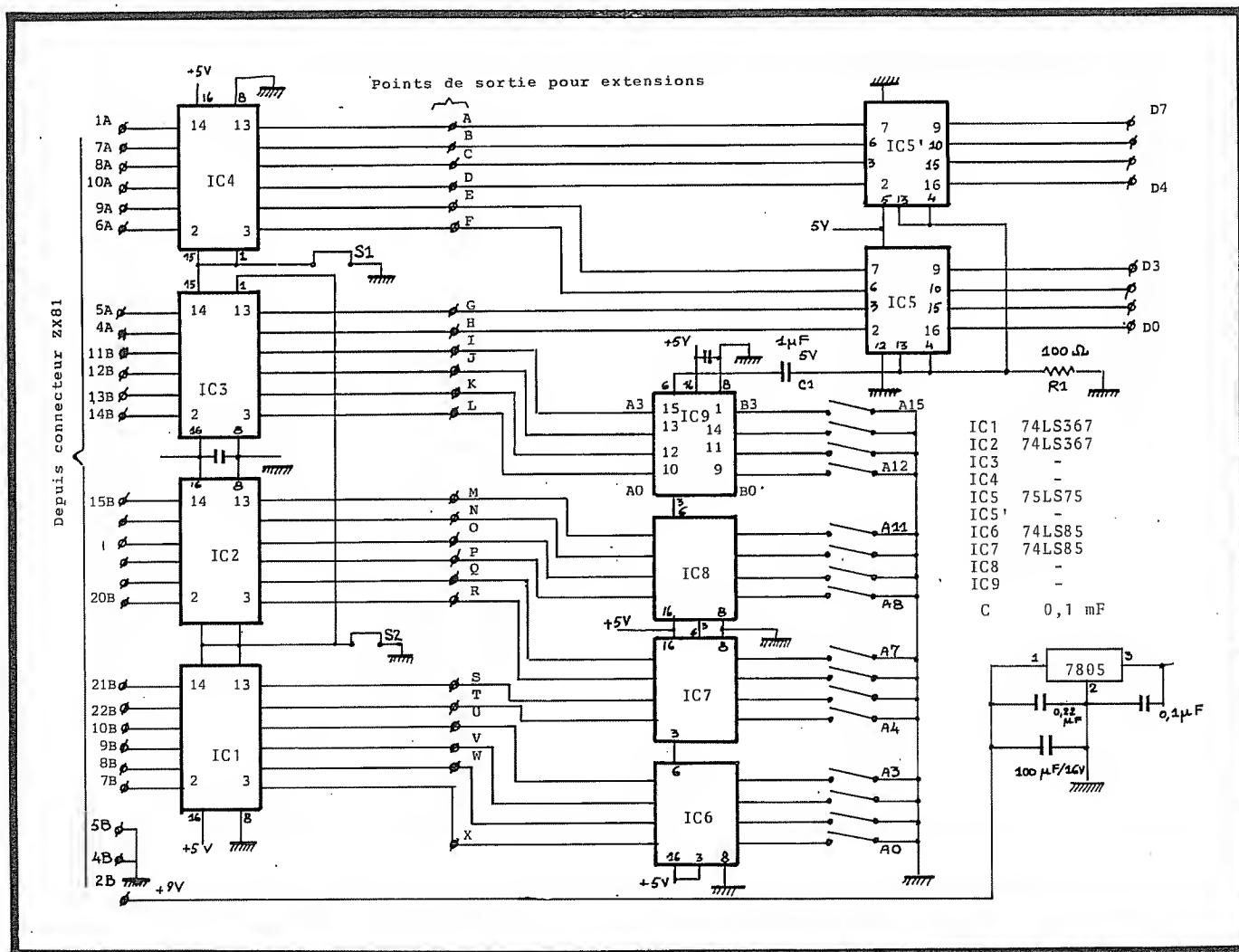
LD B,255
LD A,B
LD(16417),A
DJNZ-6
LD(16417),A
INC A
CP 255
JR C,-8
CALL $0F46
JRC,-21
RET

```



On supprime la ligne 60 du programme et en mode FAST, on fait RUN 50. On observe des signaux triangulaires de fréquence 180 Hz environ.  
Pour stopper le programme, il faut appuyer sur la

touche BREAK. Il se peut que sur certains "vieux ZX 81" on ne puisse pas interrompre le programme, il faut alors remplacer dans la liste des nombres à entrer le nombre entouré par 67.



```

0 REM GESTION SORTIES
10 DIM A$(18)
15 FOR A=1 TO 8
20 LET A$(A)="0"
30 NEXT A
40 PRINT "INTRODUIRE LE NO DE LA SORTIE"
45 INPUT A
48 IF A>8 OR A<1 THEN GOTO 40
50 PRINT "TAPER '0' OU '1'"
55 IF INKEY$="" OR INKEY$<>"0" AND INKEY$<>"1" THEN GOTO 60
70 LET A$(A)=INKEY$
80 LET F=0
90 LET D=1
100 FOR C=0 TO 1 STEP -1
110 LET D=D+VAL A$(C)*(2**F)
120 LET F=F+1
130 NEXT C
140 POKE 16417,D
150 PRINT "ETAT DES SORTIES"
160 PRINT "NO "
170 FOR X=2 TO 9
180 PRINT AT 6,X*2;X-1;AT 7,X*2;X-1
190 NEXT X
200 IF INKEY$="" THEN GOTO 200
210 GOTO 20

```

Programme 1 : Permet de changer une seule sortie à la fois en spécifiant son numéro et la valeur à mettre 0 ou 1.

```

1 REM GESTION SORTIES
2 REM EX: ADRESSE 16417
10 LET A$=""
20 PRINT "ENTRER '0' OU '1'"
30 PRINT "POUR CHAQUE SORTIE"
40 FOR S=7 TO 0 STEP -1
50 PRINT AT 10,7-S;E
55 IF INKEY$="" THEN GOTO 50
60 IF INKEY$<>"0" AND INKEY$<>"1" THEN GOTO 50
70 THEN GOTO 50
80 LET A$=A$+INKEY$
90 PRINT AT 11,0;A$
100 NEXT E
110 LET B=0
120 LET F=0
130 LET D=F
140 FOR C=B TO 1 STEP -1
150 LET D=D+VAL A$(C)*(2**F)
160 LET F=F+1
170 NEXT C
180 POKE 16417,D
190 IF INKEY$="" THEN GOTO 170
200 GOTO 10

```

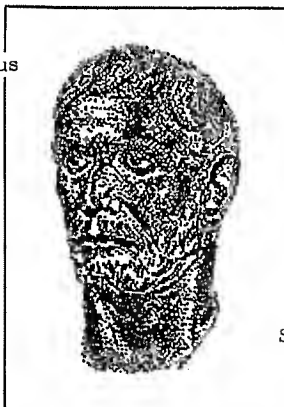
Programme 2 : Permet de changer plusieurs sorties en même temps mais oblige à réentrer la valeur de chaque sortie (0 ou 1).

G. Pedrolli

## Ecran, mon bel écran

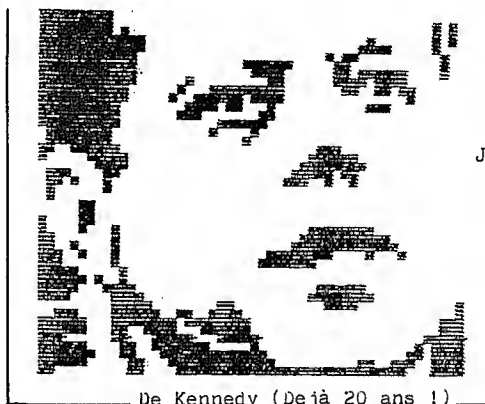
Si vous avez la chance de posséder ce bijou qu'est l'imprimante Sinclair, vous pouvez nous envoyer les plus beaux "écrans" que vous avez réalisés, quel qu'en soit le genre.

Brutus



Scherer

Navette spatiale à Cap Kennedy



De Kennedy (Déjà 20 ans !)

Jean-Luc Papillon

## Côté court

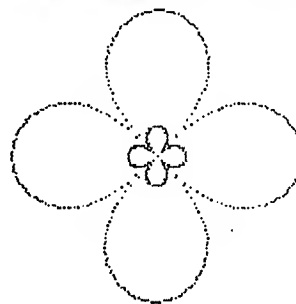
### Peinture abstraite

```
1  FOR J=1 TO 28 STEP 4
2  FOR I=1 TO 30
3  PLOT I,I#I/60+J
4  PLOT 65-I,J+I#I/60
5  NEXT I
6  NEXT J
7  PRINT AT 1,16;"■"
```

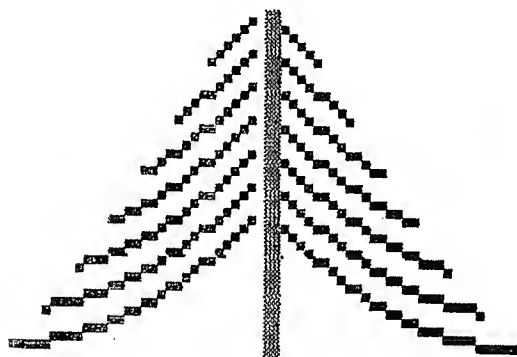
Freddie Blin

```
10 REM PEINTURE ABSTRAITE ZX81 1K
20 REM EVOLUTIVE
30 LET I=INT (RND*2)
40 LET A=INT (RND*11)
50 IF I=1 THEN LET A=A+128
60 LET X=INT (RND*32)
70 LET Y=INT (RND*32)
80 PRINT AT X,Y;CHR$ A
90 GOTO 20
```

Jean Casenave



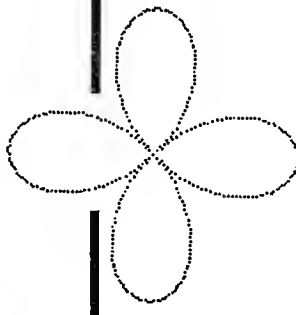
### Ah! mon beau sapin



### Roses de Noël

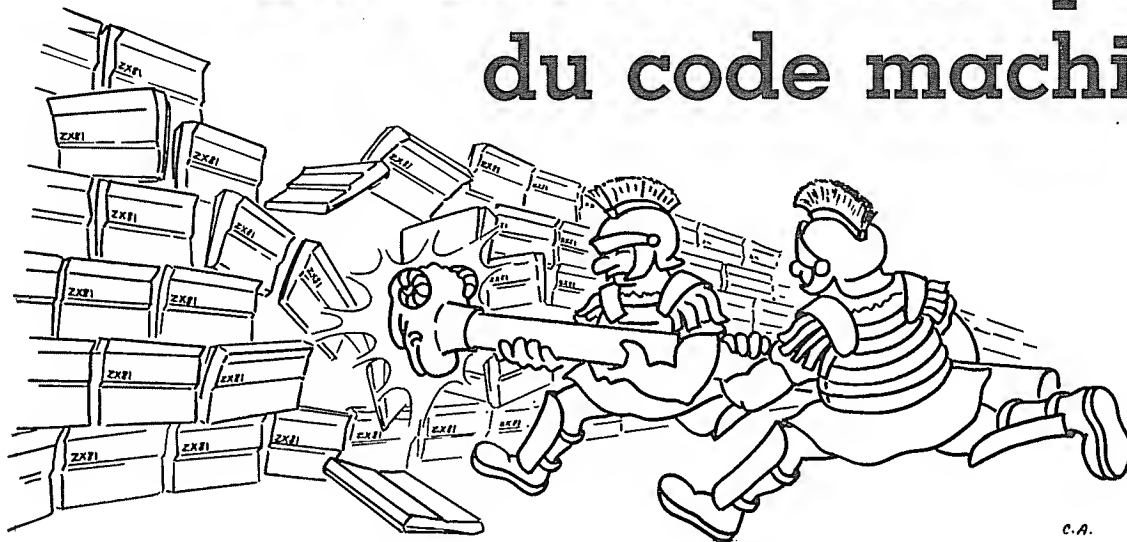
```
1 REM © MORANDINI R. & ORDI-5
2 LET C=50: LET D=50
3 FOR B=0 TO 360
4 LET A=(PI*B)/180
5 LET X=C*(50R ABS (2*COS (2*
6 A)))*COS A
7 LET Y=D*(50R ABS (2*COS (2*
8 A)))*SIN A
9 LET F=X+127: LET H=Y+88
10 PLOT F,H
11 NEXT B
12 IF C=10 THEN STOP
13 LET C=10: LET D=10
14 GO TO 10
```

Morandini



```
1 REM © MORANDINI R. & ORDI-5
2 FOR B=0 TO 360
3 LET A=(PI*B)/180
4 LET X=50*50R 2*COS (2*A)*CO
5 S A
6 LET Y=50*50R 2*COS (2*A)*SI
7 N A
8 LET F=X+127: LET H=Y+88
9 PLOT F,H
10 NEXT B
```

# Devenez un champion du code machine



Suivez pas à pas la construction d'un jeu de casse-brique particulièrement performant avec balles à effet et accroissement progressif de la vitesse d'exécution. Les erreurs d'introduction sont difficiles à envisager car codes machines et mnémoniques assembleur font l'objet de deux listages parallèles. Il est tout à fait possible d'utiliser un assembleur ZX81 sur cassette ; après le jeu de Bernard Lacoste vous trouverez une petite note sur les principales cassettes d'assembleur actuellement distribuées en France.

## UN CASSE-BRIQUE A TOUT CASSER

Ce jeu est composé de :

- 1) Une partie en BASIC de l'instruction 15 à l'instruction 240 pour :
  - dessiner le terrain
  - centraliser les différents paramètres (score, vitesse, etc...)
  - afficher les scores et gérer les parties supplémentaires.
- 2) Une partie en C.M. abritée dans les REM 1 à 5, pour le jeu lui-même ;
- 3) Les instructions 1000 à 1240 forment un programme de chargement des C.M, dérivé de celui de Marcel Henrot, pour charger des instructions en hexadécimal de un à quatre octets ;
- 4) Le programme BASIC de 2000 à la fin permet d'examiner le contenu des REM.

### La partie BASIC

Elle s'analyse sans difficulté à partir du listing joint (instructions 15 à 240).

Les variables utilisées sont :

MS : meilleur score obtenu  
 SC : score cumulé (partie en cours)  
 ST : score du tableau qui vient d'être terminé  
 V : vitesse du jeu (augmente quand V diminue)  
 DC : vitesse horizontale de la balle (± nombre de colonnes)  
 DL : vitesse verticale de la balle (± 1 ligne x 33)  
 DCN : vitesse horizontale de la balle normée (SGN(DC))

```

15 LET MS=0
17 LET SC=0
20 LET V=1000
21 POKE 16856,INT (V/256)
22 POKE 16855,V-256*INT (V/256)
23 POKE 17071,INT (V/256)
24 POKE 17070,V-256*INT (V/256)
35 CLS
40 PRINT " 000 "
50 FOR I=1 TO 3
55 PRINT " "
60 NEXT I
70 FOR I=4 TO 12
75 PRINT " "
80 NEXT I
85 FOR I=13 TO 21
90 PRINT " "
95 NEXT I
100 LET DC=INT (4*RND+.5)-2
105 LET DCN=SGN DC
110 LET DL=-33
115 POKE 16816,DC
116 POKE 16817,0
117 POKE 16818,DL
118 POKE 16819,0
119 POKE 16820,DCN
120 POKE 16821,0
121 POKE 16822,0
122 POKE 16823,0
123 POKE 16824,0
124 LET ST=USR 16801
125 LET SC=SC+ST
126 PRINT AT 15,5;"SCORE JEU ="
ST;"BRIQUES"
170 PRINT TAB 5;"SCORE PARTIE ="
SC;"BRIQUES"
175 IF MS<SC THEN LET MS=SC
180 PRINT TAB 5;"MEILLEUR SCORE ="
MS;"BRIQUES"
185 IF ST>200 THEN GOTO 210
190 PRINT "LA PARTIE EST TERMINEE."
195 PAUSE 40000
200 GOTO 17
210 PRINT "LA PARTIE CONTINUE"
211 LET V=V-100
212 IF V<100 THEN LET V=100
215 PAUSE 100
220 GOTO 21
230 SAVE "CB"
240 GOTO 0
1000 PRINT "ADRESSE DE DEBUT: ";
1005 INPUT AI
1010 PRINT AI
1015 LET NI=AI
1020 IF PEEK 16442<3 THEN SCROLL
1025 INPUT A$
1027 IF A$="S" THEN STOP
1030 IF LEN A$<2 THEN GOTO 1025
1035 IF LEN A$>2 THEN GOTO 1060
1040 POKE NI,(16*CODE A$+CODE A$(2)-476)
1045 PRINT NI;" ";A$;
1050 LET DI=1
1055 GOTO 1180

```

```

1060 IF LEN A$<4 THEN GOTO 1025
1070 IF LEN A$<4 THEN GOTO 1100
1075 POKE NI, (16*CODE A$+CODE A$
(2)-476)
1075 POKE NI+1, (16*CODE A$(3)+CO
DE A$(4)-476)
1080 PRINT NI;" ";A$(1 TO 2);" "
A$(3 TO 4);
1090 LET DI=2
1090 GOTO 1180
1100 IF LEN A$<6 THEN GOTO 1025
1105 IF LEN A$>6 THEN GOTO 1140
1110 POKE NI, (16*CODE A$+CODE A$
(2)-476)
1115 POKE NI+1, (16*CODE A$(3)+CO
DE A$(4)-476)
1120 POKE NI+2, (16*CODE A$(5)+CO
DE A$(6)-476)
1125 PRINT NI;" ";A$(1 TO 2);" "
A$(3 TO 4);" ";A$(5 TO 6);
1130 LET DI=3
1130 GOTO 1180
1140 IF LEN A$<8 THEN GOTO 1025
1145 POKE NI, (16*CODE A$+CODE A$
(2)-476)
1150 POKE NI+1, (16*CODE A$(3)+CO
DE A$(4)-476)
1155 POKE NI+2, (16*CODE A$(5)+CO
DE A$(6)-476)
1160 POKE NI+3, (16*CODE A$(7)+CO
DE A$(8)-476)

```

```

1165 PRINT NI;" ";A$(1 TO 2);" "
A$(3 TO 4);" ";A$(5 TO 6);" ";A$
(7 TO 8);
1170 LET DI=4
1180 PRINT " C""EST OK ? "
1185 INPUT A$
1190 IF A$<>" " THEN GOTO 1230
1195 LET AI=NI
1200 LET NI=AI+DI
1205 GOTO 1020
1230 PRINT AT (23-PEEK 16442),0;
"ALORS ON RECOMMENCE"
1235 PRINT AT (22-PEEK 16442),31
" "
1240 GOTO 1025
1250 STOP
1255 INPUT A
1260 CLS
1265 FOR I=0 TO 21
1270 LET V=PEEK A
1275 PRINT A;TAB 10;V;TAB 16;CHR
(V)
1280 LET A=A+1
1285 NEXT I
1290 PAUSE 40000
1295 SCROLL
1300 LET V=PEEK A
1305 PRINT A;TAB 10;V;TAB 16;CHR
(V)
1310 LET A=A+1
1315 GOTO 1250

```

## Le programme C.M.

CODES	DE	16514	A	16635	223
16514	7	00	00	00	00
16519	00	00	00	00	00
16524	00	00	00	00	00
16529	00	00	00	00	00
16534	00	00	00	00	00
16539	00	00	00	00	00
16544	00	00	00	00	00
16549	00	00	00	00	00
16554	00	00	00	00	00
16559	00	00	00	00	00
16564	00	00	00	00	00
16569	00	00	00	00	00
16574	00	00	00	00	00
16579	00	00	00	00	00
16584	00	00	00	00	00
16589	00	00	00	00	00
16594	00	00	00	00	00
16599	00	00	00	00	00
16604	00	00	00	00	00
16609	00	00	00	00	00
16614	00	00	00	00	00
16619	00	00	00	00	00
16624	00	00	00	00	00
16629	00	00	00	00	00
16634	00	00	00	00	00

CODES	DE	16642	A	16741	01
16642	42	00	00	00	00
16647	42	00	00	00	00
16652	42	00	00	00	00
16657	42	00	00	00	00
16662	42	00	00	00	00
16667	42	00	00	00	00
16672	42	00	00	00	00
16677	42	00	00	00	00
16682	42	00	00	00	00
16687	42	00	00	00	00
16692	42	00	00	00	00
16697	42	00	00	00	00
16702	42	00	00	00	00
16707	42	00	00	00	00
16712	42	00	00	00	00
16717	42	00	00	00	00
16722	42	00	00	00	00
16727	42	00	00	00	00
16732	42	00	00	00	00
16737	42	00	00	00	00

CODES	DE	16801	A	16895	195
16801	42	00	00	00	00
16806	42	00	00	00	00
16811	42	00	00	00	00
16816	42	00	00	00	00
16821	42	00	00	00	00
16826	42	00	00	00	00
16831	42	00	00	00	00
16836	42	00	00	00	00
16841	42	00	00	00	00
16846	42	00	00	00	00
16851	42	00	00	00	00
16856	42	00	00	00	00
16861	42	00	00	00	00
16866	42	00	00	00	00
16871	42	00	00	00	00
16876	42	00	00	00	00
16881	42	00	00	00	00
16886	42	00	00	00	00
16891	42	00	00	00	00

CODES	DE	16902	A	17003	175
16902	42	00	00	00	00
16907	42	00	00	00	00
16912	42	00	00	00	00
16917	42	00	00	00	00
16922	42	00	00	00	00
16927	42	00	00	00	00
16932	42	00	00	00	00
16937	42	00	00	00	00
16942	42	00	00	00	00
16947	42	00	00	00	00
16952	42	00	00	00	00
16957	42	00	00	00	00
16962	42	00	00	00	00
16967	42	00	00	00	00
16972	42	00	00	00	00
16977	42	00	00	00	00
16982	42	00	00	00	00
16987	42	00	00	00	00
16992	42	00	00	00	00
16997	42	00	00	00	00

CODES	DE	17010	A	17145	175
17010	42	00	00	00	00
17015	42	00	00	00	00
17020	42	00	00	00	00
17025	42	00	00	00	00
17030	42	00	00	00	00
17035	42	00	00	00	00
17040	42	00	00	00	00
17045	42	00	00	00	00
17050	42	00	00	00	00
17055	42	00	00	00	00
17060	42	00	00	00	00
17065	42	00	00	00	00
17070	42	00	00	00	00
17075	42	00	00	00	00
17080	42	00	00	00	00
17085	42	00	00	00	00
17090	42	00	00	00	00
17095	42	00	00	00	00
17100	42	00	00	00	00
17105	42	00	00	00	00
17110	42	00	00	00	00
17115	42	00	00	00	00
17120	42	00	00	00	00
17125	42	00	00	00	00
17130	42	00	00	00	00
17135	42	00	00	00	00
17140	42	00	00	00	00
17145	42	00	00	00	00



>DISASSEMBLE £16536

```
4099 00000000 XOR A,HL
409A 00000000 RET (HL)
409B 00000000 LD A,B
409C 00000000 ADD A,C
409D 00000000 LD A,B
409E 00000000 LD A,B
409F 00000000 LD A,B
40A0 00000000 LD A,B
40A1 00000000 LD A,B
40A2 00000000 LD A,B
40A3 00000000 LD A,B
40A4 00000000 LD A,B
40A5 00000000 LD A,B
40A6 00000000 LD A,B
40A7 00000000 LD A,B
40A8 00000000 LD A,B
40A9 00000000 LD A,B
40AA 00000000 LD A,B
40AB 00000000 LD A,B
40AC 00000000 LD A,B
40AD 00000000 LD A,B
40AE 00000000 LD A,B
40AF 00000000 LD A,B
40B0 00000000 LD A,B
40B1 00000000 LD A,B
40B2 00000000 LD A,B
40B3 00000000 LD A,B
40B4 00000000 LD A,B
40B5 00000000 LD A,B
40B6 00000000 LD A,B
40B7 00000000 LD A,B
40B8 00000000 LD A,B
40B9 00000000 LD A,B
40BA 00000000 LD A,B
40BB 00000000 LD A,B
40BC 00000000 LD A,B
40BD 00000000 LD A,B
40BE 00000000 LD A,B
40BF 00000000 LD A,B
40C0 00000000 LD A,B
40C1 00000000 LD A,B
40C2 00000000 LD A,B
40C3 00000000 LD A,B
40C4 00000000 LD A,B
40C5 00000000 LD A,B
40C6 00000000 LD A,B
40C7 00000000 LD A,B
40C8 00000000 LD A,B
40C9 00000000 LD A,B
40CA 00000000 LD A,B
40CB 00000000 LD A,B
40CC 00000000 LD A,B
40CD 00000000 LD A,B
40CE 00000000 LD A,B
40CF 00000000 LD A,B
40D0 00000000 LD A,B
40D1 00000000 LD A,B
40D2 00000000 LD A,B
40D3 00000000 LD A,B
40D4 00000000 LD A,B
40D5 00000000 LD A,B
40D6 00000000 LD A,B
40D7 00000000 LD A,B
40D8 00000000 LD A,B
40D9 00000000 LD A,B
40DA 00000000 LD A,B
40DB 00000000 LD A,B
40DC 00000000 LD A,B
40DD 00000000 LD A,B
40DE 00000000 LD A,B
40DF 00000000 LD A,B
40E0 00000000 LD A,B
40E1 00000000 LD A,B
40E2 00000000 LD A,B
40E3 00000000 LD A,B
40E4 00000000 LD A,B
40E5 00000000 LD A,B
40E6 00000000 LD A,B
40E7 00000000 LD A,B
40E8 00000000 LD A,B
40E9 00000000 LD A,B
40EA 00000000 LD A,B
40EB 00000000 LD A,B
40EC 00000000 LD A,B
40ED 00000000 LD A,B
40EE 00000000 LD A,B
40EF 00000000 LD A,B
40F0 00000000 LD A,B
40F1 00000000 LD A,B
40F2 00000000 LD A,B
40F3 00000000 LD A,B
40F4 00000000 LD A,B
40F5 00000000 LD A,B
40F6 00000000 LD A,B
40F7 00000000 LD A,B
40F8 00000000 LD A,B
40F9 00000000 LD A,B
40FA 00000000 LD A,B
40FB 00000000 LD A,B
40FC 00000000 LD A,B
40FD 00000000 LD A,B
40FE 00000000 LD A,B
40FF 00000000 LD A,B
```

>DISASSEMBLE £16565

```
40AB 00000000 LD A,B
40AC 00000000 LD A,B
40AD 00000000 LD A,B
40AE 00000000 LD A,B
40AF 00000000 LD A,B
40B0 00000000 LD A,B
40B1 00000000 LD A,B
40B2 00000000 LD A,B
40B3 00000000 LD A,B
40B4 00000000 LD A,B
40B5 00000000 LD A,B
40B6 00000000 LD A,B
40B7 00000000 LD A,B
40B8 00000000 LD A,B
40B9 00000000 LD A,B
40BA 00000000 LD A,B
40BB 00000000 LD A,B
40BC 00000000 LD A,B
40BD 00000000 LD A,B
40BE 00000000 LD A,B
40BF 00000000 LD A,B
40C0 00000000 LD A,B
40C1 00000000 LD A,B
40C2 00000000 LD A,B
40C3 00000000 LD A,B
40C4 00000000 LD A,B
40C5 00000000 LD A,B
40C6 00000000 LD A,B
40C7 00000000 LD A,B
40C8 00000000 LD A,B
40C9 00000000 LD A,B
40CA 00000000 LD A,B
40CB 00000000 LD A,B
40CC 00000000 LD A,B
40CD 00000000 LD A,B
40CE 00000000 LD A,B
40CF 00000000 LD A,B
40D0 00000000 LD A,B
40D1 00000000 LD A,B
40D2 00000000 LD A,B
40D3 00000000 LD A,B
40D4 00000000 LD A,B
40D5 00000000 LD A,B
40D6 00000000 LD A,B
40D7 00000000 LD A,B
40D8 00000000 LD A,B
40D9 00000000 LD A,B
40DA 00000000 LD A,B
40DB 00000000 LD A,B
40DC 00000000 LD A,B
40DD 00000000 LD A,B
40DE 00000000 LD A,B
40DF 00000000 LD A,B
40E0 00000000 LD A,B
40E1 00000000 LD A,B
40E2 00000000 LD A,B
40E3 00000000 LD A,B
40E4 00000000 LD A,B
40E5 00000000 LD A,B
40E6 00000000 LD A,B
40E7 00000000 LD A,B
40E8 00000000 LD A,B
40E9 00000000 LD A,B
40EA 00000000 LD A,B
40EB 00000000 LD A,B
40EC 00000000 LD A,B
40ED 00000000 LD A,B
40EE 00000000 LD A,B
40EF 00000000 LD A,B
40F0 00000000 LD A,B
40F1 00000000 LD A,B
40F2 00000000 LD A,B
40F3 00000000 LD A,B
40F4 00000000 LD A,B
40F5 00000000 LD A,B
40F6 00000000 LD A,B
40F7 00000000 LD A,B
40F8 00000000 LD A,B
40F9 00000000 LD A,B
40FA 00000000 LD A,B
40FB 00000000 LD A,B
40FC 00000000 LD A,B
40FD 00000000 LD A,B
40FE 00000000 LD A,B
40FF 00000000 LD A,B
```

>DISASSEMBLE £16577

```
40C1 00000000 CALL Z,40E7
40C2 00000000 LD A,B
40C3 00000000 LD A,B
40C4 00000000 LD A,B
40C5 00000000 LD A,B
40C6 00000000 LD A,B
40C7 00000000 LD A,B
40C8 00000000 LD A,B
40C9 00000000 LD A,B
40CA 00000000 LD A,B
40CB 00000000 LD A,B
40CC 00000000 LD A,B
40CD 00000000 LD A,B
40CE 00000000 LD A,B
40CF 00000000 LD A,B
40D0 00000000 LD A,B
40D1 00000000 LD A,B
40D2 00000000 LD A,B
40D3 00000000 LD A,B
40D4 00000000 LD A,B
40D5 00000000 LD A,B
40D6 00000000 LD A,B
40D7 00000000 LD A,B
40D8 00000000 LD A,B
40D9 00000000 LD A,B
40DA 00000000 LD A,B
40DB 00000000 LD A,B
40DC 00000000 LD A,B
40DD 00000000 LD A,B
40DE 00000000 LD A,B
40DF 00000000 LD A,B
40E0 00000000 LD A,B
40E1 00000000 LD A,B
40E2 00000000 LD A,B
40E3 00000000 LD A,B
40E4 00000000 LD A,B
40E5 00000000 LD A,B
40E6 00000000 LD A,B
40E7 00000000 LD A,B
40E8 00000000 LD A,B
40E9 00000000 LD A,B
40EA 00000000 LD A,B
40EB 00000000 LD A,B
40EC 00000000 LD A,B
40ED 00000000 LD A,B
40EE 00000000 LD A,B
40EF 00000000 LD A,B
40F0 00000000 LD A,B
40F1 00000000 LD A,B
40F2 00000000 LD A,B
40F3 00000000 LD A,B
40F4 00000000 LD A,B
40F5 00000000 LD A,B
40F6 00000000 LD A,B
40F7 00000000 LD A,B
40F8 00000000 LD A,B
40F9 00000000 LD A,B
40FA 00000000 LD A,B
40FB 00000000 LD A,B
40FC 00000000 LD A,B
40FD 00000000 LD A,B
40FE 00000000 LD A,B
40FF 00000000 LD A,B
```

>DISASSEMBLE £16602

```
40D1 00000000 LD A,B
40D2 00000000 LD A,B
40D3 00000000 LD A,B
40D4 00000000 LD A,B
40D5 00000000 LD A,B
40D6 00000000 LD A,B
40D7 00000000 LD A,B
40D8 00000000 LD A,B
40D9 00000000 LD A,B
40DA 00000000 LD A,B
40DB 00000000 LD A,B
40DC 00000000 LD A,B
40DD 00000000 LD A,B
40DE 00000000 LD A,B
40DF 00000000 LD A,B
40E0 00000000 LD A,B
40E1 00000000 LD A,B
40E2 00000000 LD A,B
40E3 00000000 LD A,B
40E4 00000000 LD A,B
40E5 00000000 LD A,B
40E6 00000000 LD A,B
40E7 00000000 LD A,B
40E8 00000000 LD A,B
40E9 00000000 LD A,B
40EA 00000000 LD A,B
40EB 00000000 LD A,B
40EC 00000000 LD A,B
40ED 00000000 LD A,B
40EE 00000000 LD A,B
40EF 00000000 LD A,B
40F0 00000000 LD A,B
40F1 00000000 LD A,B
40F2 00000000 LD A,B
40F3 00000000 LD A,B
40F4 00000000 LD A,B
40F5 00000000 LD A,B
40F6 00000000 LD A,B
40F7 00000000 LD A,B
40F8 00000000 LD A,B
40F9 00000000 LD A,B
40FA 00000000 LD A,B
40FB 00000000 LD A,B
40FC 00000000 LD A,B
40FD 00000000 LD A,B
40FE 00000000 LD A,B
40FF 00000000 LD A,B
```

>DISASSEMBLE £16615

```
40F1 00000000 LD A,B
40F2 00000000 LD A,B
40F3 00000000 LD A,B
40F4 00000000 LD A,B
40F5 00000000 LD A,B
40F6 00000000 LD A,B
40F7 00000000 LD A,B
40F8 00000000 LD A,B
40F9 00000000 LD A,B
40FA 00000000 LD A,B
40FB 00000000 LD A,B
40FC 00000000 LD A,B
40FD 00000000 LD A,B
40FE 00000000 LD A,B
40FF 00000000 LD A,B
```

>DISASSEMBLE £16642

```
4102 00000000 LD HL,(4082)
4103 00000000 LD DE,(4084)
4104 00000000 ADD HL,DE
4105 00000000 BIT 7,E
4106 00000000 JR Z,410F
4107 00000000 DEC H
4108 00000000 LD DE,(4086)
4109 00000000 ADD HL,DE
410A 00000000 BIT 7,E
410B 00000000 JR Z,4119
410C 00000000 DEC H
410D 00000000 RET
410E 00000000 RET
```

>DISASSEMBLE £16668

```
411C 00000000 LD A,(4086)
411D 00000000 NEG A
411E 00000000 LD (4086),A
411F 00000000 RET
4120 00000000 ADD A,B
4121 00000000 ADD A,B
4122 00000000 LD A,(408A)
4123 00000000 NEG A
4124 00000000 LD (408A),A
4125 00000000 LD A,(4084)
4126 00000000 NEG A
4127 00000000 LD (4084),A
4128 00000000 RET
4129 00000000 ADD A,B
```

>DISASSEMBLE £16698

```
413A 00000000 CALL 4127
413B 00000000 CALL 411C
413C 00000000 RET
413D 00000000 ADD A,B
413E 00000000 ADD A,B
413F 00000000 DEC HL
4140 00000000 LD A,H
4141 00000000 OR L
4142 00000000 JR NZ,4143
4143 00000000 RET
4144 00000000 ADD A,B
4145 00000000 ADD A,B
4146 00000000 LD HL,(4091)
4147 00000000 INC HL
```

>DISASSEMBLE £16719

```
414F 00000000 LD (4091),HL
4150 00000000 LD HL,(408C)
4151 00000000 INC HL
4152 00000000 INC HL
4153 00000000 INC HL
4154 00000000 INC HL
4155 00000000 LD A,(HL)
4156 00000000 OR A,B
4157 00000000 JR NZ,4158
4158 00000000 LD (HL),0C
4159 00000000 DEC HL
415A 00000000 JR 18F6
415B 00000000 INC HL
415C 00000000 LD (HL),A
415D 00000000 LD HL,(408C)
```

>DISASSEMBLE £16741

```
4168 00000000 RET
4169 00000000 NOP
416A 00000000 DEC E
416B 00000000 LD E,1F
416C 00000000 JR NZ,416D
416D 00000000 LD (4163),HL
416E 00000000 DEC H
416F 00000000 NOP
4170 00000000 DEC E
4171 00000000 LD E,1F
4172 00000000 LD E,1F
```

>DISASSEMBLE £16801

```
41A1 00000000 LD HL,(408C)
41A2 00000000 LD BC,0204
41A3 00000000 ADD HL,BC
41A4 00000000 LD (4088),HL
41A5 00000000 LD (HL),03
41A6 00000000 INC HL
41A7 00000000 LD (HL),03
41A8 00000000 INC HL
41A9 00000000 LD (HL),03
41AA 00000000 LD HL,0000
41AB 00000000 NOP
41AC 00000000 NOP
41AD 00000000 LD A,(4090)
41AE 00000000 OR 01
41AF 00000000 RET
```

>DISASSEMBLE £16827

```
41B8 00000000 JR C,41C3
41B9 00000000 DEC A
41BA 00000000 LD (4090),A
41BB 00000000 JR 41C8
41BC 00000000 LD BC,(4081)
41BD 00000000 RET
41BE 00000000 LD HL,(4088)
41BF 00000000 LD BC,0030
41C0 00000000 XOR A
41C1 00000000 SBC HL,BC
```

```
41D1 00000000 LD (HL),54
41D2 00000000 LD (4082),HL
41D3 00000000 LD HL,03E8
41D4 00000000 CALL 4143
```

>DISASSEMBLE £16850

```
41D0 00000000 LD HL,(4025)
41D1 00000000 LD A,DF
41D2 00000000 OR H
41D3 00000000 JR NZ,41EB
41D4 00000000 LD A,7F
41D5 00000000 OR L
41D6 00000000 JR NZ,41EB
41D7 00000000 RST 08
41D8 00000000 LD (822A),HL
41D9 00000000 LD B,B
41DA 00000000 LD BC,0001
41DB 00000000 LD DE,(4086)
41DC 00000000 ADD HL,DE
41DD 00000000 BIT 7,E
```

>DISASSEMBLE £16888

```
41F8 00000000 JR Z,41FB
41F9 00000000 DEC H
41FA 00000000 CALL 4095
41FB 00000000 JR 4205
```



Le déplacement de la balle, pour chaque cycle, est défini par deux valeurs :

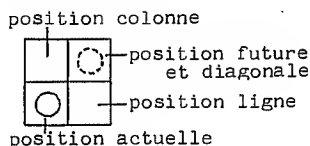
DC, vitesse horizontale : nombre de colonnes parcourues,

DL, vitesse verticale : nombre de lignes parcourues x 33.

Si on appelle A l'adresse actuelle de la balle, l'adresse future sera  $A' = A + DC + DL$ .

Avant chaque déplacement de la balle, on regarde quels obstacles sont touchés dans le sens de son mouvement.

Exemple : DC = +1  
DL = -1



Si les obstacles touchés forment une ligne horizontale, le rebond se fera en chargeant le signe de DL. Si les obstacles touchés forment une ligne verticale, le rebond se fera en chargeant le signe de DC. Si les obstacles forment un coin, rentrant ou saillant, la balle reviendra sur ses pas.

Les briques touchées sont effacées et le compteur de score incrémenté, mais tout caractère autre qu'un espace sera considéré comme un obstacle, donc produira un rebond. Si de plus le caractère "touché" correspond à la raquette, la vitesse horizontale est incrémentée ou décrémentée de 1 colonne selon le sens de déplacement de celle-ci.

Si l'adresse de la balle correspond à une ligne située sous la raquette, elle est considérée comme sortie, s'efface et une nouvelle balle est produite. Après la 5ème balle, le jeu est terminé et rend le contrôle au BASIC.

La raquette se déplace si les touches "1" ou "0" sont reconnues. Afin de laisser quelques chances au joueur, elle se déplace 2 fois plus vite que la balle.

Adresse	C.M.	Mnémonique
TABLE DES REM		
1 REM	122 car.	de 16514 à 16635
2 REM	153 car.	de 16642 à 16794
3 REM	95 car.	de 16801 à 16895
4 REM	102 car.	de 16902 à 17003
5 REM	136 car.	de 17010 à 17145
VARIABLES ET PARAMETRES		
16514/15		Adresse de la balle
16516/17		DC, vitesse horizontale (-2, -1, 0, +1 ou +2)
16518/19		DL, vitesse verticale x 33 (±33)
16520/21		Adresse raquette
16522/23		DCN (= SGN DC)
16524/27		libre
16528		Balles restant à jouer
16529/30		Score du jeu en cours
16531/35		libre
16536		Début 1ère routine

#### DEMONTAGE DU PROGRAMME

- INIT, 16801/19 : La raquette est située au début au milieu de la ligne 21. L'adresse de son élément gauche est sauvegardée, puis on affiche 3 caractères graphiques (code 03).

- NOUV.BAL

. 16822 à 16839 : vérifie s'il reste au moins une balle en réserve (octet 16528). Si oui, décrémente le contenu de la réserve et passe à la suite. Si non, rend le contrôle au BASIC.

. 16840 à 16853 : calcule l'adresse de la balle pour la situer sur la raquette, et l'affiche, et sauvegarde son adresse.

- MV.BAL

. 16854/59 : règle la rapidité du jeu par la routine DELAI, qui crée une pause dont la durée dépend de HL.

. 16860/74 : permet l'interruption du programme si

la touche "B" est reconnue.

. 16875/93 : après avoir initialisé B à 0 et C à 1, calcule l'adresse de la position colonne en ajoutant DL à l'adresse de la balle, et appelle la routine TEST pour la position colonne.

. 16894/95 : passage au REM suivant.

. 16902/909 : évite de tester les positions ligne et diagonale pour un déplacement vertical (DC donc le registre E est alors à 0).

- D.RAQ :

. 17069/74 : petite pause avant chaque déplacement de la raquette, pour éviter un trop grand effet de saccade.

. 17075/90 : à partir de l'adresse, recalcule la position de l'élément gauche de la raquette sur sa ligne.

. 17091/94 : limite le déplacement à gauche.

. 17096/17119 : déplace la raquette d'une colonne à gauche si la touche "1" est reconnue.

. 17120/145 : idem p. déplacement à droite (butée et touche "0").

- SUITE

. 16935/77 : modifie la vitesse de la balle selon la valeur du registre B, qui dépend de la configuration des obstacles rencontrés :

B = 0 : pas d'obstacles

B ≠ 0 (1 à 7) : changement du signe de DL, DC ou des deux par appel aux routines  $\overline{DL}$ ,  $\overline{DC}$  ou  $\overline{DL\overline{C}}$ .

- DEP.BAL.

. 16978/89 : calcule la nouvelle adresse de la balle compte tenu des composantes de sa vitesse après rebond. Pour éviter des effacements intempestifs d'obstacles (pour des vitesses DC > 1), la vitesse DC est normée si nécessaire.

. 16990/17001 : déplace effectivement la balle

. 17002/3 : passe au REM suivant.

- SORTIE

. 17010/28 : vérifie si l'adresse de la balle est ou non au-dessous de la ligne 21. Si non passe à MV.RAQ.

Si oui, la balle est sortie. Dans ce cas :

. 17029/65 : efface la balle, et après une attente redéfinit la nouvelle vitesse de la balle avec un certain élément de hasard selon la position de la raquette, puis revient à NV.BAL. (nouvelle balle).

- MV.RAQ. appelle 2 fois D.RAQ. avant de sauter à MV.BAL. En pokant 1, 2 ou 4 à l'adresse 17060, la raquette se déplacera 1, 2 ou 4 fois plus vite que la balle

. 16910/20 : calcule l'adresse de la position diagonale avant d'appeler TEST. C est chargé de la valeur 2.

. 16921/34 : idem pour la position ligne, mais avec la valeur 4 pour C.

- Routine TEST

. 16536/41

a) revient au programme principal si le contenu de l'adresse testée est nul (pas d'obstacle).

b) ajoute C à B dans le cas contraire. Ainsi après avoir testé les 3 positions, B contiendra une valeur de 0 à 7 qui permettra de définir le type de rebond à effectuer.

. 16542/66 : dans ce cas, la balle touche un obstacle. Ce peut être une brique, le bord du terrain ou la raquette.

S'il s'agit d'une brique, elle est effacée et le score est incrémenté par appel à la routine COMPT. Sinon, s'il s'agit à la fois d'une position colonne et de la raquette, on continue sur EFFET.

- EFFET, 16567/80 : reconnaît les touches 1 et 0 et fait appel si nécessaire à EFG ou EFD (effet à gauche ou à droite).

- EFG, 16593/16614 : décrémente la vitesse horizontale d'une unité : si la balle va vers la gauche, elle ira plus vite, si elle va à droite, elle ira moins vite.

- EFD, 16815/35 : idem pour l'effet à droite.

- NORM : ramène à ±1 la vitesse horizontale.

B. Lacoste

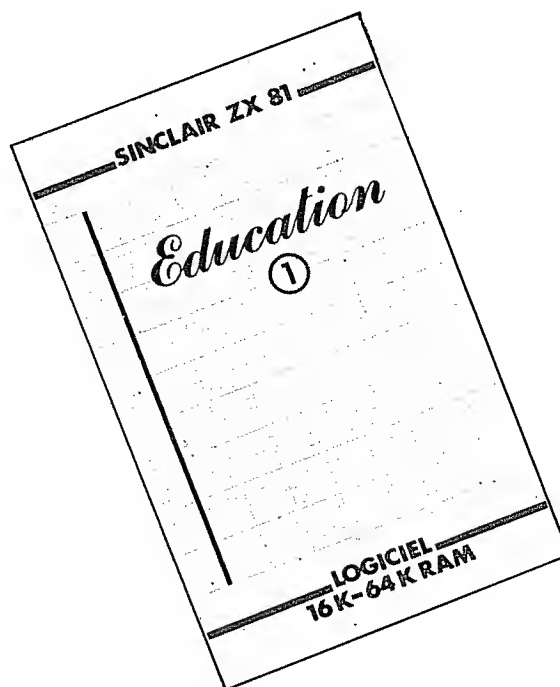
## Trois utilitaires ZX81

ORDI-5 vous donne rapidement son avis sur trois cassettes utilitaires destinées aux programmeurs confirmés qui font honneur à ce qu'on appelle, dans les milieux technocratiques "le génie logiciel" français.

EDUCATION SINCLAIR 16K-64K DE RAM 100 F

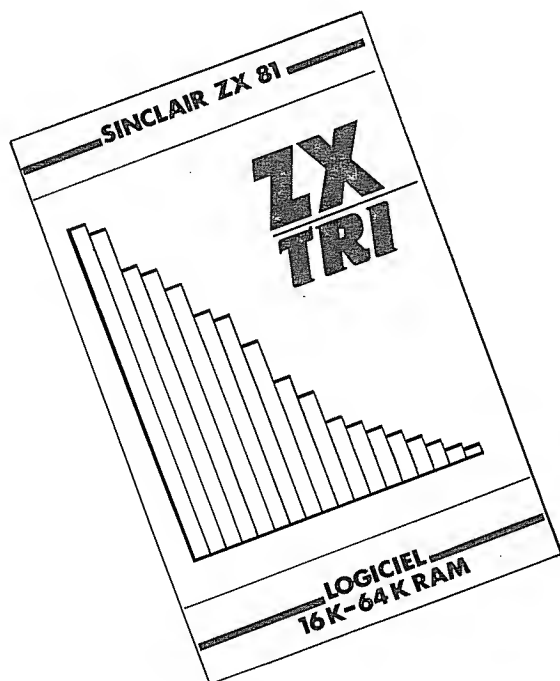
Voici une cassette assez originale dans son objet et à laquelle nous ne connaissons pas d'équivalent sur le marché. Beaucoup d'enseignants ou de formateurs sont très intéressés par les possibilités pédagogiques des petits ordinateurs surtout lorsqu'ils ne sont pas très coûteux et peuvent donc être fournis en plusieurs exemplaires à une classe plus ou moins nombreuse. Malheureusement la construction d'un didacticiel, c'est-à-dire d'un programme orienté sur la pédagogie, implique que son auteur soit un programmeur confirmé afin de tirer le meilleur parti de l'organisation mémoire et trouver les procédures les plus souples ; enfin il est nécessaire que la programmation soit structurée pour éviter tout plantage de l'élève.

La cassette "éducation" de Sinclair vient à l'aide des formateurs qui souhaitent construire des didacticiels sans pour cela programmer superbement en BASIC. Cette cassette est en quelque sorte une aide à la construction de didacticiels. Elle demande à l'utilisateur, que la documentation appelle très justement l'"initiateur", quel est le type d'exercice qu'il veut programmer, comment vont se répartir les différents textes, les écrans, quel est le mode choisi (questions-réponses) et le logiciel se structure lui-même en fonction des options choisies.



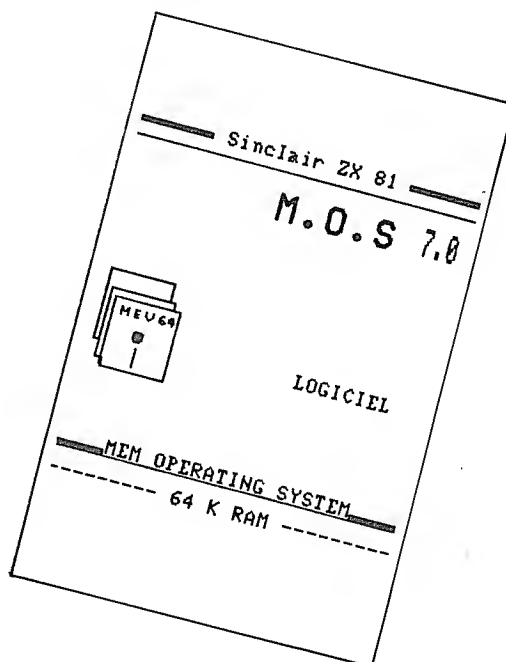
Ce programme a une portée très générale et la possibilité de l'adapter au matériel choisi 16 ou 64 K lui donne encore un attrait supplémentaire. Il est en particulier, dans la version 64K, possible d'arriver à des compositions d'écran très élaborées. Le seul inconvénient est que la démarche, lourde et rigide, implique une grande attention de la part de l'initiateur et probablement un bon nombre de tâtonnements pour découvrir la meilleure formule, d'autant que la notice est assez dense et assez peu vivante (c'est évidemment inévitable lorsqu'il faut faire état d'un grand nombre de possibilités) : de toutes façons, à partir du moment où l'on se lance dans la construction de didacticiels, c'est que l'on a pris une option sur des applications sérieuses et que par conséquent on est prêt à travailler. Il ne faut pas se laisser rebuter par l'aspect un peu ingrat de cette cassette au premier abord : on peut la considérer comme un utilitaire de la meilleure catégorie. Nous aimerions assez entendre parler d'expériences à l'aide de ce programme.

Nous n'avons pas peur du pléonasme, le ZX TRI est l'un des utilitaires les plus utiles que nous ayons rencontrés pour le ZX81, du moins en ce qui concerne la construction de programmes de fichier. Entendons-nous bien : le ZX TRI n'est pas un programme de fichier au sens où on l'entend habituellement. Les programmes de fichiers ordinaires présentent un certain nombre de rigidités qui limitent considérablement leur usage : qu'ils soient spécialisés ou que, d'une façon plus sophistiquée, ils proposent à l'utilisateur de définir lui-même ses propres champs, ces programmes sont intangibles : surtout écrits en langages machine, ils reposent sur une organisation stricte des différentes routines (insertions, recherche, correction) au cœur desquelles l'utilisateur ne peut pas intervenir sans tout bouleverser.



Avec le ZX TRI on a affaire à une toute autre philosophie : on attend de l'utilisateur qu'il se construise lui-même son programme de fichier, mais s'ajoutent, à la batterie des instructions BASIC habituelles, 5 mots nouveaux pratiquement utilisables comme les mots du dictionnaire de base, chacun de ces mots correspondant à une fonction extrêmement utile et puissante d'insertion, de recherche, de suppression d'un élément de tri. La simple utilisation de ces mots : INS, DEL, REC, avec des paramètres correctement choisis appelle des routines machine qui se chargent des additions, soustractions, et décalages. Ces différents sous-programmes seraient longs à écrire et surtout à exécuter en BASIC, même avec intervention sur les variables-système : l'incrémentement d'un tableau de N éléments déclarés à N + 1 éléments, en cas d'insertion par exemple, n'est pas à la portée du premier programmeur venu et nécessite en tous cas une parfaite connaissance de la disposition de la mémoire. Grâce à ces instructions, simplement introduites dans une REM et précédées d'un RAND USR 16516 il est possible de réaliser des programmes de fichier étonnamment courts et rapides et comportant l'essentiel des procédures nécessaires dans ce type d'application. Les seules difficultés, mineures, que nous ayons rencontrées concernent l'édition lorsque l'on choisit la version 64K. En définitive une belle réalisation très souple qui laisse entièrement l'initiative au programmeur.

L'opérating système d'un ordinateur est sommairement défini comme le programme en langage machine qui permet de gérer les fichiers sur disquettes. Comment le ZX81, qui ne peut être muni de disquettes (pour l'instant !) peut-il supporter un opérating système ? Il s'agit en fait plutôt d'un simulateur d'opérating système sur mémoire vive : en travaillant sous ce logiciel, l'utilisateur a l'impression de travailler comme avec une disquette ; il peut ainsi écrire différents programmes même rédigés avec des numéros de lignes identiques et les envoyer dans la mémoire sans se préoccuper de l'endroit où ils seront stockés, le logiciel le fera pour lui. L'ensemble des commandes que l'on trouve pour la gestion de disquettes et ainsi présent : FORMAT, pour préparer la mémoire à recevoir des programmes calibrés, CAT pour lire instantanément l'ensemble des fichiers de la mémoire, etc. Cet o.s. permet de faire des chaînages entre différents programmes, de les renumérotés, de les détruire sélectivement, etc. C'est une grande satisfaction de pouvoir travailler sous une dizaine de programmes différents, et même plus, sans avoir à passer et repasser par les opérations fastidieuses de sauvegarde et chargement cassettes.



Les va-et-vient se font instantanément. La programmation est confortable et on se sent vraiment travailler "sous" O.S.

Ce logiciel n'occupe que 2K de mémoire, ce qui permet de la placer en différents endroits et donc de le rendre compatible avec d'autres softs performants (FLM ou ZX TRI). Nous ne voyons pas d'autre alternative pour ceux qui souhaitent travailler en ayant une quinzaine de programmes à leur disposition immédiate. Cette belle réalisation souffre de petites limitations que nous nous n'avons pas sû contourner facilement : si l'on veut chaîner des programmes il faut les avoir tous directement transcrit au clavier ; l'O.S. ne gère que la zone programme les variables ne peuvent passer de programme à programme qu'à l'aide des artifices habituels.

B. T.

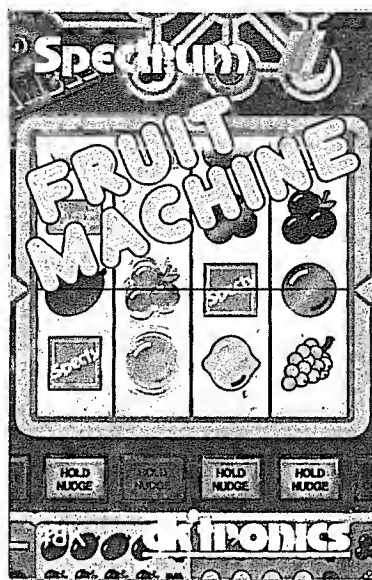
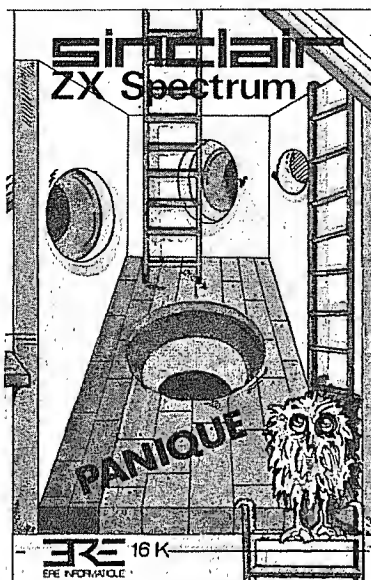


## 5 Cinq cassettes spectrum

PANIQUE (ERE INFORMATIQUE 1983) 16 K OU 48 K - 75 F

Ere Informatique a réussi là un joli jeu très coloré et au graphisme parfaitement clair. Les auteurs annoncent qu'il s'agit d'un jeu d'arcade et effectivement il n'est pas sans rappeler le jeu bien connu dans les cafés où les gorilles précipitent des tonneaux sur des bonhommes qui disposent des échelles, etc...

Le propos est plus familier puisque tout se passe de la cave au grenier dans une maison et qu'il s'agit simplement d'éviter qu'un animal qui ressemble à une grosse chouette ne mange une provision de choux. Le nombre de décisions à prendre simultanément est important et, spécialement pour ceux qui n'ont pas de manette, il est difficile de gérer les touches curseur simultanément pour lever une échelle, percer un trou ou aller et venir. Jeu difficile mais franchement amusant, cependant l'absence de variété se fait sentir à la longue. Rapidité moyenne.

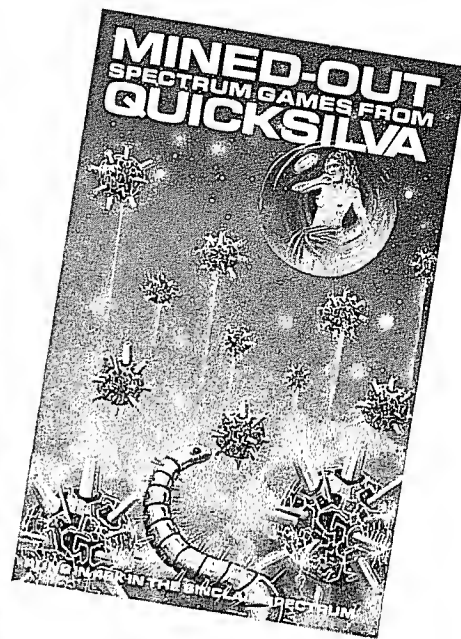


FRUIT MACHINE (DK TRONICS 1983) 16 OU 48 K - 75 F

L'un des plus beaux jeux de chez DK Tronics : un graphisme réellement époustoufflant pour servir un jeu de Jack-Pot assez classique. Il faut noter la très grande rapidité dans le défilement des figures (oranges, citrons, etc) dont le réalisme en dit long sur les étonnantes possibilités du Spectrum dans le do-

MINED OUT - (QUICKSILVA 83) - 16 OU 48 K - 80 F

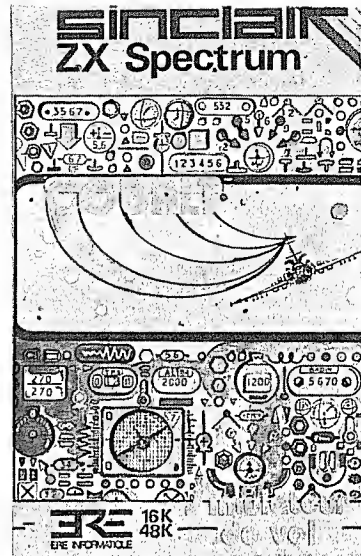
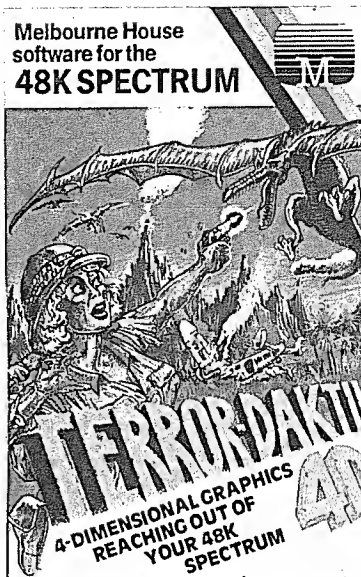
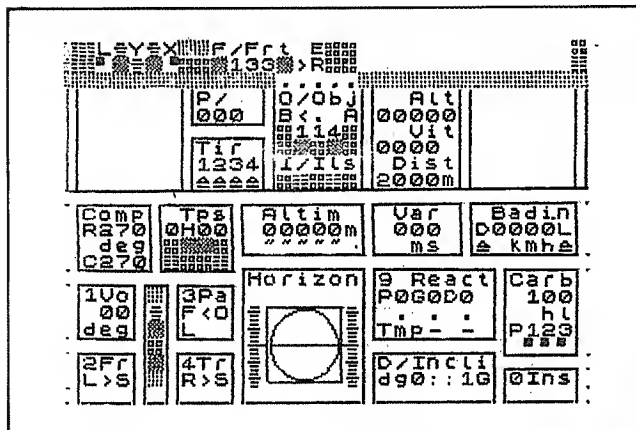
Un jeu dont on se lasse difficilement bien que le principe en soit très simple : il faut progresser dans un champ de mines dont on ne connaît pas l'emplacement pour atteindre l'autre extrémité du champ. On peut être distrait de sa route par quelques demoiselles éplorées. Un indicateur de proximité précise le nombre de mines qui menacent directement et il faut alors revenir sur ses pas pour repartir dans une autre direction. Les niveaux se succèdent en difficulté. Une fois la passe terminée, le cheminement suivi se reconstitue, cette fois avec les mines en évidence. Il s'agit d'une belle réalisation en BASIC de rapidité moyenne mais d'un réel intérêt.



maine graphique. Nous sommes restés très admiratifs devant la musique d'accompagnement spirituelle. C'est une véritable fête sur l'écran. Il faut s'accrocher pour savoir ce que l'on fait (montant des paris, etc.), et faire très attention à ne pas toucher "BREAK" car alors le jeu s'auto-efface. Un gros défaut toutefois, qu'il devrait être possible de corriger : les instructions du programme ne sont pas francisées, or elles sont assez complexes.

C'est une variante intéressante des fameux "invaders" qui se démarque des autres jeux de ce genre par la qualité du graphisme ambiant, ce qui change des figures géométriques de ce jeu peu trop souvent vu. L'argument est le suivant : votre avion s'écrase dans une jungle inconnue et dans ce décor hostile les envahisseurs sont des espèces (!) de ptérodactyles préhistoriques très difficiles à écarter avec un simple fusil : l'un deux finit toujours par fondre sur vous d'un seul coup, et là, c'est assez effrayant. Décor très riche constitué d'une montagne, de rivières, etc... et naturellement de la carcasse de l'avion ; une certaine monotonie toutefois due au jeu lui-même.

INTERCEPTEUR COBALT (ERE INFORMATIQUE) 16 OU 48 K - 95F



Enfin un très beau jeu interactif complet entièrement en français : près de 45 commandes rendent ce simulateur de vol particulièrement performant et même un modèle du genre visiblement très documenté. Ce type de jeu requiert énormément de concentration et la faculté de pouvoir fixer très attentivement un tableau. Nous reproduisons ci-dessous un bel écran.

Ph. D.

## L'ordinateur en classe

# Le ZX fort en thème

L'ordinateur fort en thème ? Ou encore, quel est le génitif de Zédix ? Zedivis ? Zedinis ? Soyons sérieux, avec J.C. Hein qui nous propose d'avoir un pied dans le passé et un pied dans l'avenir avec ce beau programme de déclinaison latine.

```

1 REM *****
2 J. C. HEIN
3 (C) 1983
4 *****
5 REM
6 GOSUB 9000
7 GOTO 12
8 STOP
9 CLS
10 REM DECLINAISONS LATINES
11 LET ZX=0
12 LET PA=0
13 PRINT "DECLINAISONS LA
14 TINES"
15 PRINT
16 PRINT
17 PRINT "NOMINATIF DU MOT ?"
18 INPUT N$
19 IF LEN N$ < 3 AND N$ <> "OS" TH
20 EN GOTO 60
21 PRINT N$
22 PRINT
23 PRINT "GENITIF DU MOT ?"
24 INPUT Q$
25 IF LEN Q$ < 3 THEN GOTO 83
26 PRINT Q$
27 LET A$=Q$(1 TO LEN Q$-2)
28 LET Q$=Q$(LEN Q$-1 TO )
29 IF Q$="AE" OR Q$="IS" OR Q$
30 ="UI" OR Q$="EI" THEN GOTO 120
31 LET A$=Q$(1 TO LEN Q$-1)
32 LET Q$=Q$(LEN Q$ TO )
33 IF Q$ <> "I" THEN GOTO 10

```

```

118 REM *****
119 IF Q$="ITINERIS" THEN GOTO
120 584
121 IF Q$="HOMINIS" THEN GOTO 5
122 59
123 IF Q$="PATRIS" OR Q$="MATRI
124 S" OR Q$="FRATRIS" OR Q$="JUVENI
125 S" OR Q$="GENIS" OR Q$="CANIS" T
126 HEN GOTO 560
127 IF Q$="AE" THEN GOTO 130
128 IF Q$="I" THEN GOTO 227
129 IF Q$="IS" THEN GOTO 310
130 IF Q$="US" THEN GOTO 550
131 IF Q$="EI" THEN GOTO 720
132 REM *****
133 REM *****PREMIERE*****
134 LET A$="A AE A A
135 E AM AS AE ARUM AE
136 IS A IS " ARUM AE
137 CLS
138 PRINT TAB 5;"PREMIERE DECLI
139 NAISON"
140 PRINT
141 LET V$=N$
142 GOTO 1000
143 REM *****
144 REM *****DEUXIEME*****
145 LET A$="US I E I
146 UM OS I ORUM O I
147 IS O IS " ARUM AE
148 IF N$((LEN N$-1 TO )="UM"
149 THEN GOTO 250
150 IF N$((LEN N$ TO )="R" THE
151 N GOTO 275
152 231 CLS

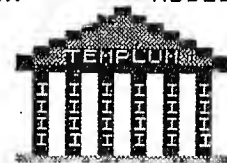
```



```
232 PRINT TAB 5;"DEUXIEME DECLI
NATION"
234 GOTO 1000
250 LET R$="UM I A UM A
UM O A IS " ORUM O A
IS
255 CLS
260 PRINT " DEUXIEME DECLINAISO
(NEUTRE)"
265 PRINT
270 GOTO 1000
275 LET R$="O$ ( TO (LEN O$-1))
277 LET ZX=1
280 LET A$="
UM I I
UM O OS IS " ORUM O I
IS
285 CLS
290 PRINT TAB 5;"DEUXIEME DECLI
NATION"
295 PRINT
300 GOTO 1000
305 REM *****
310 REM *****TROISIEME*****
320 REM PARISYLLABIQUE ?
340 IF N$(1 TO (LEN R$))<>R$ TH
EN GOTO 344
342 GOTO 350
344 LET PA=0
346 GOTO 380
350 LET A$=N$((LEN R$+1) TO )
352 IF LEN A$=0 OR LEN A$>2 THE
N GOTO 356
354 GOTO 360
356 LET PA=0
360 GOTO 380
365 IF A$="E" OR A$="ES" OR A$=
"O" THEN LET PA=1
370 REM -----FAUX IMPAIRS-----
380 IF LEN R$<2 THEN GOTO 384
382 GOTO 390
384 LET FP=0
386 GOTO 420
390 LET H$=R$(LEN R$-1)
392 LET L$=R$(LEN R$)
394 LET FP=0
400 IF H$="A" OR H$="E" OR H$="
O" OR H$="U" OR H$="Y"
THEN GOTO 420
410 IF L$<>"A" AND L$<>"E" AND
L$<>"I" AND L$<>"O" AND L$<>"U"
AND L$<>"Y" THEN LET FP=1
415 REM GENRE?
420 PRINT
425 PRINT "GENRE DU MOT ?"
427 PRINT
428 PRINT "
-MASCULIN:1
-FEMININ:2
-NEUTRE:3
"
429 PRINT "ENTREZ LE NUMERO COR
RESPONDANT."
430 INPUT Z$
435 IF Z$<>"1" AND Z$<>"2" AND
Z$<>"3" THEN GOTO 430
440 REM FAUX IMPARISYLLABIQUES
442 IF Z$="3" AND (N$(LEN N$-1
) TO )="AR" OR N$(LEN N$-1) TO
)="AL" THEN LET PA=1
460 REM DESINENCES
480 IF Z$="3" AND PA=1 THEN GOT
O 495
492 GOTO 510
495 LET A$="E IA E I
E IA IS " IUM I I
IBUS I IBUS " IUM I I
IBUS
500 CLS
502 LET ZX=2
504 PRINT " 3 EME:PARISYLLABIQU
E NEUTRES"
506 PRINT
508 GOTO 1000
510 REM PARI+FAUX PARI
520 IF FP=1 OR PA=1 THEN GOTO 5
522 GOTO 540
525 LET A$="IS ES IS E
S EM ES IS " IUM I E
IBUS E IBUS " IUM I E
IBUS
530 CLS
532 PRINT
535 PRINT " 3 EME:PARI. ET FAU
X IMPARI."
537 GOTO 1000
540 REM IMPARISYLLAB. NEUTRE
550 IF Z$="3" THEN GOTO 554
552 GOTO 560
554 LET A$="A IS UM I A
IBUS E IBUS " UM I A
IBUS
556 LET ZX=2
558 PRINT "3 EME:IMPARISYLLABIQ
UE NEUTRES"
560 PRINT
562 GOTO 1000
565 LET A$="ES ES IS E
S EM ES IS " UM I E
IBUS E IBUS " UM I E
IBUS
568 LET ZX=1
570 CLS
572 PRINT " 3 EME:IMPARISYLL
ABIQUES"
575 PRINT
580 GOTO 1000
585 REM *****
590 REM *****QUATRIEME*****
595 LET A$="US US US U
UM US US " UUM UI
IBUS U IBUS " UUM UI
IBUS
598 IF N$(LEN N$)="U" THEN GOTO
700
600 CLS
605 PRINT TAB 5;"QUATRIEME DECL
INAISON"
610 PRINT
```

```
695 GOTO 1000
700 LET A$="U UA U U
A U UA US " UUM UI
IBUS U IBUS " UUM UI
IBUS
705 CLS
710 PRINT " QUATRIEME DECLINAIS
ON (NEUTRE)"
713 PRINT
717 GOTO 1000
720 REM *****
725 REM *****CINQUIEME*****
730 LET A$="ES ES E
S EM ES EI ES " ERUM EI
EBUS E EBUS " ERUM EI
EBUS
740 CLS
750 PRINT TAB 5;"CINQUIEME DECL
INAISON"
760 PRINT
765 REM
770 REM
775 REM *****
780 REM *****
785 REM *****
790 PRINT "CAS: SINGULIER: PL
URIEL:"
1020 PRINT "
1030 LET X=1
1035 LET Y=1
1040 LET W$="NOM.VOC.ACC.GEN.DAT
.ABL."
1050 FOR D=6 TO 16 STEP 2
1060 PRINT AT D,0;W$(X TO X+3)
1070 LET X=X+4
1075 PRINT AT D,6;R$+A$(Y TO Y+5
)
1080 LET Y=Y+6
1085 IF ZX=1 OR ZX=2 THEN PRINT
AT 6,6;N$;"AT 8,6;N$;"
1087 IF ZX=2 THEN PRINT AT 10,6;
N$;"
1090 PRINT AT D,18;R$+A$(Y TO Y+
5)
1100 LET Y=Y+6
1110 NEXT D
1113 LET ZX=0
1115 REM *****
1116 REM *****
1120 PRINT AT 20,0;"VOULEZ-VOUS
RECOMMENCER ? (O/N)"
1130 INPUT S$
1140 IF S$<>"O" AND S$<>"N" THEN
GOTO 1130
1150 IF S$="N" THEN GOTO 9
1155 GOTO 10
9000 PRINT AT 5,5;"
9010 PRINT TAB 5;"
9020 PRINT TAB 5;"
9030 PRINT TAB 5;"
9040 PRINT TAB 5;"
9050 PRINT TAB 5;"
9070 PRINT AT 12,16;"
9075 PRINT AT 13,14;"
9080 PRINT AT 14,12;"
9085 PRINT AT 15,10;"
9100 PRINT AT 16,10;"
9110 PRINT AT 17,11;"
9112 PRINT AT 18,11;"
9114 PRINT AT 19,11;"
9116 PRINT AT 20,11;"
9118 PRINT AT 21,10;"
9200 FOR N=0 TO 60
9210 NEXT N
9220 RETURN
```

PREMIERE DECLINAISON		
CAS:	SINGULIER:	PLURIEL:
NOM.	ROSA	ROSAE
VOC.	ROSA	ROSAE
ACC.	ROSAM	ROSAS
GEN.	ROSAE	ROSARUM
DAT.	ROSAE	ROSIS
ABL.	ROSA	ROSIS



J.C. Hein

# Programmes de jeu

BLINDES (ZX81 16K)

Nous allons sacrifier dans ce numéro à la vogue des "wargames". Dans cette bataille de chars, qui se déroule très lentement, il faut anticiper les mouvements de l'adversaire et donner des instructions chiffrées à ses propres unités : attention, un "wargame" n'est pas un jeu interactif, on est plutôt comme dans un état-major à voir évoluer des maquettes sur une grande carte ; ce jeu implique une certaine capacité de concentration et d'abstraction.

```

1 RAND
2 CLEAR
3 CLS
10 PRINT AT 0,11;"KOBRA"
20 PRINT AT 1,11;" "
30 PRINT AT 2,11;" "
35 PRINT AT 3,11;" "
40 PRINT AT 4,11;" "
45 PRINT AT 5,11;" "
50 PRINT AT 6,11;" "
55 PRINT AT 7,11;" "
60 PRINT AT 8,11;" "
65 PRINT AT 9,11;" "
70 PRINT AT 10,11;" "
75 PRINT AT 11,11;" "
100 FOR Z=1 TO 29
110 PRINT AT 13,8;"CORRECTIONS"
120 PRINT AT 13,8;"COMBAT"
125 PRINT AT 16,Z;" "
130 PRINT AT 17,Z;" "
135 PRINT AT 20,6;"COPYRIGHT 83"
140 NEXT Z
142 FOR A=12 TO 21 STEP .25
143 PRINT AT A,14;" "
145 PRINT AT A,14;" "
147 NEXT A
148 PAUSE 300
150 CLS
160 PRINT AT 6,0;" "
465 PRINT AT 7,0;"CANONS (SYM"
470 PRINT AT 8,0;"BOLE: UN NUMERO)"
475 PRINT AT 9,0;"(NOMBRE MAX"
480 PRINT AT 10,0;"IMUH : 5)"
482 PRINT AT 11,0;"BOUCLERS"
484 PRINT AT 12,0;"SYMBOLE : ---"
510 PRINT AT 21,2;"NB DE BOUCLER"
520 INPUT NB
525 IF NB>3 OR NB<0 THEN GOTO 5
530 CLS
535 PRINT AT 4,0;"ENTREZ LE PO"
540 PRINT AT 5,0;"ALLESSE PROUV"
545 PRINT AT 6,0;"ATTENTION XON"
546 PRINT AT 9,0;"RENTREZ LE"
550 INPUT PB
555 IF PB>14 OR PB<0 THEN GOTO 5
560 LET PBZ=INT (RAND*17)+13
565 CLS
570 PRINT AT 20,PB;" "
575 PRINT AT 0,PBZ;"0"
580 PRINT AT 1,PBZ-2;"-----"
585 FOR A=0 TO 30
590 PRINT AT INT (RAND*3)+9,A;" "
595 PRINT AT INT (RAND*3)+10,A;" "
600 PRINT AT INT (RAND*3)+8,A;" "
605 NEXT A
610 FOR M=1 TO INT (RAND*7)+1
615 PRINT AT INT (RAND*11)+1,INT"
620 (RAND*31);" "
625 PRINT AT INT (RAND*11)+9,INT"
630 (RAND*31);" "
635 NEXT M
640 PRINT AT 21,0;"ACCEPTEZ-VOUS"
645 PRINT AT 21,0;"ATTENTION: NE PAS"
1000 IF INKEY$="" THEN GOTO 1000

```

```

1001 IF INKEY$="" THEN GOTO 100
1002 IF INKEY$<>"0" THEN GOTO 94
1004 PRINT AT 21,0;"POS. DES CAN"
1005 DIM Z(5)
1010 DIM U(5)
1020 FOR C=1 TO 5
1025 INPUT Z(C)
1030 IF Z(C)<15 OR Z(C)>20 THEN
GOTO 1025
1035 INPUT U(C)
1040 IF U(C)>31 OR U(C)<0 THEN G
OTO 1030
1040 PRINT AT Z(C),U(C);CHR$ (C+
165)
1050 NEXT C
1060 PRINT AT 21,0;"CORRECTION"
1065 IF INKEY$="" THEN GOTO 1065
1070 IF INKEY$="0" THEN GOTO 110
1080 GOTO 1190
1100 PRINT AT 21,0;"NUMERO DU CA"
1110 INPUT U
1111 PRINT AT Z(U),U(U);" "
1120 LET U1=U
1130 PRINT AT 21,0;"NOUVELLES PO"
1140 INPUT Z(U1)
1145 IF Z(U1)<12 OR Z(U1)>20 THE
N GOTO 1140
1150 INPUT U(U1)
1155 IF U(U1)>31 OR U(U1)<0 THEN
GOTO 1150
1160 PRINT AT Z(U1),U(U1);CHR$ (
U+165)
1170 GOTO 1060
1190 IF NB=0 THEN GOTO 1410
1200 PRINT AT 21,0;"ENTREZ LES P"
1205 DES BOUCLERS
1210 DIM N(NB)
1220 DIM M(NB)
1230 FOR B=1 TO NB
1240 INPUT N(B)
1245 IF N(B)<12 OR N(B)>20 THEN
GOTO 1240
1250 INPUT M(B)
1255 IF M(B)>28 OR M(B)<0 THEN G
OTO 1250
1260 PRINT AT N(B),M(B);"--";B;"
"
1265 NEXT B
1270 PRINT AT 21,0;"CORRECTIONS"
1280 IF INKEY$="" THEN GOTO 1280
1290 IF INKEY$="0" THEN GOTO 130
0
1295 GOTO 1400
1300 PRINT AT 21,0;"NUMERO DU BO"
1310 INPUT U
1320 PRINT AT N(U),M(U);" "
1330 PRINT AT 21,0;"NOUVELLES PO"
1340 LET U1=U
1350 INPUT N(U1)
1355 IF N(U1)<12 OR N(U1)>20 THE
N GOTO 1350
1360 INPUT M(U1)
1365 IF M(U1)>28 OR M(U1)<0 THEN
GOTO 1360
1370 PRINT AT N(U1),M(U1);"--";U
;"
1380 GOTO 1270
1400 PRINT AT 21,0;"LE PLACE"
1410 LET PP=PEEK 16396+PEEK 1639
7*255+1
1415 PRINT AT 6,0;" " ; AT 6,31;" "
1425 DIM E(3)
1430 DIM F(3)
1435 LET E(1)=7
1440 LET F(1)=5
1445 LET E(2)=7
1450 LET F(2)=15
1455 LET E(3)=7
1460 LET F(3)=25
1470 PRINT AT E(1),F(1);"C";AT E
(2),F(2);"C";AT E(3),F(3);"C"
1500 IF NB=0 THEN GOTO 1580
1510 DIM G(NB)
1520 DIM H(NB)
1525 FOR J=1 TO NB
1530 LET G(J)=INT (RAND*7)+1
1540 LET H(J)=INT (RAND*28)
1545 IF PEEK (PP+(G(J)*33)+H(J))
=CODE "X" THEN GOTO 1530
1550 IF PEEK (PP+(G(J)*33)+H(J))
=CODE "C" THEN GOTO 1530
1560 PRINT AT G(J),H(J);"-----"
1570 NEXT J
1580 PRINT AT 21,0;"LES CHARS SE"
1582 PAUSE 120

```

```

1565 DIM Y(5)
1590 DIM X(5)
1600 LET Y(1)=20
1610 LET X(1)=15
1620 LET Y(2)=17
1630 LET X(2)=19
1640 LET Y(3)=20
1650 LET X(3)=20
1660 LET Y(4)=21
1670 LET X(4)=20
1680 LET Y(5)=23
1690 LET X(5)=20
1700 PRINT AT Y(1),X(1),"1",AT Y
(2),X(2),"2",AT Y(3),X(3),"3",AT
Y(4),X(4),"4",AT Y(5),X(5),"5"
1705 DIM D(6)
1706 DIM O(6)
1710 LET D(1)=2
1720 LET O(1)=2
1730 LET D(2)=4
1740 LET O(2)=4
1750 LET D(3)=5
1760 LET O(3)=5
1770 LET D(4)=6
1780 LET O(4)=6
1790 LET D(5)=10
1800 LET O(5)=10
1805 LET D(6)=12
1807 LET O(6)=12
1810 PRINT AT O(1),D(1),"2",AT O
(2),D(2),"3",AT O(3),D(3),"5",AT
O(4),D(4),"6",AT O(5),D(5),"8",
AT O(6),D(6),"10"
1820 PRINT AT 21,0;"VOUS ENGAGEZ
VOTRE COMBAT"
1830 FOR A=1 TO 40
1840 NEXT A
1852 IF PEEK (PP+660+PB) <> CODE "
" THEN GOTO 9993
1855 PRINT AT 21,0;"VOUS ENGAGEZ
VOTRE COMBAT"
1860 IF INKEY$="" THEN GOTO 1860
1870 IF INKEY$="C" THEN GOTO 190
1880 IF INKEY$="D" THEN GOTO 300
1900 IF INKEY$="P" THEN GOTO 900
1905 IF PEEK (PP+(Y(1)*33)+X(1))
<> CODE "1" AND PEEK (PP+(Y(2)*33
)+X(2)) <> CODE "2" AND PEEK (PP+(Y
(3)*33)+X(3)) <> CODE "3" AND PEE
K (PP+(Y(4)*33)+X(4)) <> CODE "4"
AND PEEK (PP+(Y(5)*33)+X(5)) <> CO
DE "5" THEN GOTO 7450
1910 PRINT AT 21,0;"NUMERO DU
CHAR JEU ?"
1920 INPUT NU
1930 IF NU>5 OR NU<=0 THEN GOTO
1910
2020 IF PEEK (PP+(Y(NU)*33)+X(NU
)) <> CODE CHR$ (NU+28) THEN GOTO
7000
2025 PRINT AT 21,0;"REPLACEMENT
DE L'OUTRER (F) ?"
2026 IF INKEY$="" THEN GOTO 2026
2030 IF INKEY$="F" THEN GOTO 225
2035 IF INKEY$="K" THEN GOTO 204
2037 GOTO 2020
2040 PRINT AT 21,0;"DIRECTION DE
REPLACEMENT ?"
2050 INPUT DA
2055 LET NA=INT (RND*5)+1
2057 PRINT AT Y(NU),X(NU)," "
2060 FOR A=1 TO NA
2062 LET VAR=INT (RND*5)
2065 LET YA2=((SIN ((2*PI)/
360)))*A)
2066 LET XA2=((COS (DA*((2*PI)/3
60)))*A)
2067 IF PEEK ((PP+(YA2+Y(NU))*33
+(XA2+X(NU))))=CODE "X" THEN GOT
O 7000
2068 IF PEEK ((PP+(YA2+Y(NU))*33
+(XA2+X(NU))))=CODE "X" THEN GOT
O 7000
2069 IF PEEK ((PP+(YA2+Y(NU))*33
+(XA2+X(NU))))=CODE "-" THEN GOT
O 8990
2070 IF VAR=0 OR VAR=4 THEN GOTO
5000
2072 IF YA2+Y(NU)>20 OR YA2+Y(NU
)<0 THEN GOTO 2086
2077 PRINT AT INT (YA2+Y(NU)),IN
T (XA2+X(NU));CHR$ (NU+28)
2080 PRINT AT INT (YA2+Y(NU)),IN
T (XA2+X(NU));" "
2086 NEXT A
2090 LET Y(NU)=INT (Y(NU)+YA2)
2092 LET X(NU)=INT (X(NU)+XA2)
2098 PRINT AT Y(NU),X(NU);CHR$ (
NU+28)
2099 GOTO 9000
2250 PRINT AT 21,0;"DIRECTION DU
CHAR (EN DEGRE) ?"
2260 INPUT FA
2270 FOR A=1 TO 5
2275 LET FXA=INT ((COS (FA*((2*P
I)/360)))*A)
2280 LET FYA=INT ((SIN (-FA*((2*
PI)/360)))*A)
2300 IF (FYA+Y(NU))>20 OR (FYA+Y
(NU))<0 THEN GOTO 2360

```

```

2350 PRINT AT (FYA+Y(NU)),(FXA+X
(NU));" "
2357 PRINT AT (FYA+Y(NU)),(FXA+X
(NU));" "
2360 NEXT A
2370 GOTO 9000
2380 PRINT AT 6,0;"3",AT 6,31;"2"
5005 PRINT AT 6,1;"++++++"
5010 PRINT AT 6,1;" "
5015 IF Y(NU)=6 THEN GOTO 7000
5020 GOTO 2077
5099 IF PEEK (PP+(O(L)*33)+D(L))
<> CODE CHR$ (L+156) THEN GOTO 90
20
5000 IF PEEK (PP+(Y(P)*33)+X(P))
<> CODE CHR$ (P+26) THEN GOTO 902
5
5001 LET I=O(L)
5002 LET O=D(L)
5003 FOR S=1 TO 5
5010 LET D1=(SGN (Y(P)-I))
5020 LET D2=(SGN (X(P)-O))
5025 LET I=I+D1
5030 LET O=O+D2
5033 PRINT AT I,0;" "
5035 PRINT AT I,0;" "
5040 NEXT S
5042 GOTO 1850
5500 IF PEEK (PP+(O(L)*33)+D(L))
<> CODE CHR$ (L+156) THEN GOTO 90
30
5510 IF PEEK (PP+(Z(P)*33)+U(P))
<> CODE CHR$ (P+165) THEN GOTO 90
30
5520 LET I1=O(L)
5530 LET O1=D(L)
5540 FOR S=1 TO 5
5545 LET D3=(SGN (Z(P)-I1))
5550 LET D4=(SGN (U(P)-O1))
5555 LET I1=I1+D3
5560 LET O1=O1+D4
5565 PRINT AT I1,O1;" "
5570 PRINT AT I1,O1;" "
5580 NEXT S
5585 GOTO 1850
5600 IF PEEK (PP+(O(L)*33)+D(L))
<> CODE CHR$ (L+156) THEN GOTO 90
20
5610 LET I3=O(L)
5620 LET O3=D(L)
5630 FOR S=1 TO 5
5635 LET D5=(SGN (20-I3))
5640 LET D6=(SGN (PB-O3))
5645 LET I3=I3+D5
5650 LET O3=O3+D6
5655 PRINT AT I3,O3;" "
5660 PRINT AT I3,O3;" "
5670 NEXT S
5710 GOTO 1850
7000 PRINT AT 21,0;" LE CHAR ";N
U;" A EXPLOSE."
7010 POKE (PP+(Y(NU)*33)+X(NU)),
0
7015 PAUSE 250
7020 GOTO 1850
7450 PRINT AT 21,0;"VOUS VAS CHA
RISSENT DETAIL ?"
7460 PAUSE 250
7470 GOTO 1850
8020 PRINT AT 21,0;"NUMERO DU CH
AR JEU ?"
8030 INPUT CC
8032 IF CC>5 OR CC<=0 THEN GOTO
8020
8035 IF PEEK (PP+(Z(CC)*33)+U(CC
)) <> CODE CHR$ (CC+165) THEN GOTO
8150
8040 PRINT AT 21,0;"DIRECTION DE
L'OUTRER ?"
8045 INPUT FC
8050 FOR C=1 TO 5
8055 LET FYC=INT ((SIN ((-FC*((2*
PI)/360)))*C)
8065 LET FXC=INT ((COS (FC*((2*P
I)/360)))*C)
8070 IF (Z(CC)+FYC)>20 OR (Z(CC)
+FYC)<0 THEN GOTO 8120
8100 PRINT AT Z(CC)+FYC,W(CC)+FX
C;" "
8110 PRINT AT Z(CC)+FYC,W(CC)+FX
C;" "
8120 NEXT C
8130 GOTO 9000
8150 PRINT AT 21,0;"LE CANONNE
MENT DETAIL ?"
8160 PAUSE 250
8170 GOTO 1850
8990 PRINT AT 21,0;"BOULDER
EXPLOSION ?"
8992 PAUSE 250
9000 IF PEEK (PP+PBZ) <> CODE "0"
THEN GOTO 9800
9003 PRINT AT 21,0;"*****"
9010 PRINT AT 0,PBZ;"0"
9015 LET HA=INT (RND*6)+1
9018 IF PEEK (PP+(O(HA)*33)+D(HA
)) <> CODE CHR$ (HA+156) THEN GOTO
9000
9021 IF Y(1)<=12 AND Y(1)>=2 OR
Y(2)<=12 AND Y(2)>=2 OR Y(3)<=12
AND Y(3)>=2 OR Y(4)<=12 AND Y(4)

```



```

) >=2 OR Y(5) <=12 AND Y(5) >=2 THE
N GOSUB 9200
90222 FOR L=1 TO 5
90223 FOR P=1 TO 5
90224 IF ABS (O(L)-20) <=5 AND ABS
(D(L)-P) <=5 THEN GOTO 6600
90225 IF ABS (O(L)-Y(P)) <=5 AND A
BS (D(L)-X(P)) <=5 THEN GOTO 5999
90226 IF ABS (O(L)-Z(P)) <=5 AND A
BS (D(L)-U(P)) <=5 THEN GOTO 6500
90230 NEXT P
90235 NEXT L
90250 LET OO=INT (RND*5)+1
90260 PRINT AT O(HA),D(HA); " "
90265 FOR A=1 TO OO
90270 IF PEEK (PP+(Y(1)*33)+X(1))
<>CODE "1" AND PEEK (PP+(Y(2)*33
)+X(2)) <>CODE "2" AND PEEK (PP+(
Y(3)*33)+X(3)) <>CODE "3" AND PEE
K (PP+(Y(4)*33)+X(4)) <>CODE "4"
AND PEEK (PP+(Y(5)*33)+X(5)) <>CO
DE "5" THEN GOTO 9990
90280 LET DD1=(SGN (Y(INT (RND*5)
)+1)-O(HA))+(A-1)
90285 LET DD2=(SGN (X(INT (RND*5)
)+1)-D(HA))+(A-1)
90290 IF PEEK (PP+((O(HA)+DD1)*33
)+(D(HA)+DD2))=CODE "C" THEN GOS
UB 9600
9100 IF PEEK (PP+((O(HA)+DD1)*33
)+(D(HA)+DD2))=CODE "X" THEN GOT
O 9500
9105 IF (O(HA)+DD1) >20 THEN LET
O(HA)=0
9106 IF (O(HA)+DD1) <0 THEN LET O
(HA)=20
9107 IF (D(HA)+DD2) >31 THEN LET
D(HA)=0
9108 IF (D(HA)+DD2) <0 THEN LET D
(HA)=31
9110 PRINT AT O(HA)+DD1,D(HA)+DD
1;CHR$(HA+156)
9120 PRINT AT O(HA)+DD1,D(HA)+DD
1;CHR$ 0
9125 NEXT A
9130 LET D(HA)=D(HA)+DD2
9140 LET O(HA)=O(HA)+DD1
9150 IF O(HA)=5 THEN GOTO 9033
9155 POKE (PP+(O(HA)*33)+D(HA)),
CODE CHR$(HA+156)
9160 GOTO 1850
9170 POKE (PP+((O(HA)+1)*33)+D(H
A)), CODE CHR$(HA+156)
9180 GOTO 1850
9200 FOR I=1 TO 3
9205 IF PEEK (PP+(E(I)*33)+F(I))
<>CODE "C" THEN GOTO 9250
9210 FOR V=1 TO INT (RND*5)+1
9211 IF PEEK (PP+(E(I)*33)+F(I)+
V)=CODE "A" OR PEEK (PP+(E(I)*33
)+F(I)+V)=CODE "B" OR PEEK (PP+(
E(I)*33)+F(I)+V)=CODE "C" OR PEE
K (PP+(E(I)*33)+F(I)+V)=CODE "D"
OR PEEK (PP+(E(I)*33)+F(I)+V)=C
ODE "E" OR PEEK (PP+(E(I)*33)+F(
I)+V)=CODE "F" THEN GOTO 9221
9215 PRINT AT E(I),F(I)+V; " "
9220 PRINT AT E(I),F(I)+V; " "
9221 IF PEEK (PP+(E(I)*33)+F(I)-
V)=CODE "A" OR PEEK (PP+(E(I)*33
)+F(I)-V)=CODE "B" OR PEEK (PP+(
E(I)*33)+F(I)-V)=CODE "C" OR PEE
K (PP+(E(I)*33)+F(I)-V)=CODE "D"
OR PEEK (PP+(E(I)*33)+F(I)-V)=C
ODE "E" OR PEEK (PP+(E(I)*33)+F(
I)-V)=CODE "F" THEN GOTO 9227
9222 PRINT AT E(I),F(I)-V; " "
9225 PRINT AT E(I),F(I)-V; " "
9227 IF PEEK (PP+(E(I)+V)*33)+F
(I))=CODE "A" OR PEEK (PP+(E(I)
)+V)*33)+F(I))=CODE "B" OR PEEK (
PP+(E(I)+V)*33)+F(I))=CODE "C"
OR PEEK (PP+(E(I)+V)*33)+F(I))=
CODE "D" OR PEEK (PP+(E(I)+V)*3
3)+F(I))=CODE "E" OR PEEK (PP+(
E(I)+V)*33)+F(I))=CODE "F" THEN
GOTO 9233
9230 PRINT AT E(I)+V,F(I); " "
9232 PRINT AT E(I)+V,F(I); " "
9233 IF PEEK (PP+(E(I)-V)*33)+F
(I))=CODE "A" OR PEEK (PP+(E(I)
)-V)*33)+F(I))=CODE "B" OR PEEK (
PP+(E(I)-V)*33)+F(I))=CODE "C"
OR PEEK (PP+(E(I)-V)*33)+F(I))=
CODE "D" OR PEEK (PP+(E(I)-V)*3
3)+F(I))=CODE "E" OR PEEK (PP+(
E(I)-V)*33)+F(I))=CODE "F" THEN
GOTO 9245
9235 PRINT AT E(I)-V,F(I); " "
9237 PRINT AT E(I)-V,F(I); " "
9240 NEXT V
9245 NEXT I
9250 RETURN
9260 PRINT AT 21,0;"UN DES CHIFF
RES"
9265 PAUSE 1250
9270 GOTO 1850
9280 LET DD1=-DD1
9290 LET DD2=-DD2
9300 RETURN
9310 CLS
9310 PRINT AT 8,6;"LA BASE DU ZX
A SAUTE..."
9320 PRINT AT 10,10;"BRAVO"
9330 GOTO 9995
9340 LET DD1=SGN (20-D(HA))+(A-1)

```

```

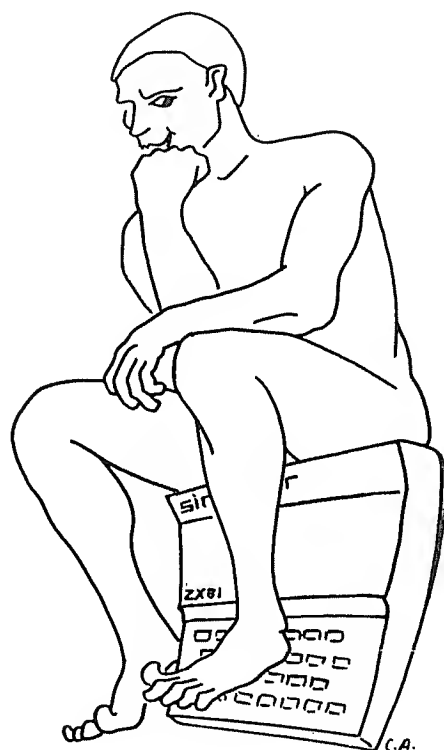
9991 LET DD2=SGN (PB-D(HA))+(A-1)
9992 GOTO 9090
9993 CLS
9994 PRINT AT 10,0;"VOTRE BASE E
ST DETRUITE..."
9995 PRINT AT 12,0;"VOUS REJOURE
SSEZ V"
9996 IF INKEY$="" THEN GOTO 9994
9998 IF INKEY$="R" THEN RUN
9999 STOP
+2,M*2+5; PAPER 2;"V"; AT N*2+2,M
*2+5; "V": NEXT W: FOR W=0 TO 60:
PRINT AT N*2,M*2+3;"V"; AT N*2,M
*2+3; PAPER 2;"V": NEXT W
9720 GO TO 235
9750 LET D=0: GO SUB 9550: IF D=
1 AND K=0 THEN GO TO 5122
9760 LET P(N,M)=0: LET P(N+1,M-1
)=0: FOR W=0 TO 60: PRINT AT N*2
+2,M*2+1; PAPER 2;"V"; AT N*2+2,M
*2+1; "V": NEXT W: FOR W=0 TO 60:
PRINT AT N*2,M*2+3; "V"; AT N*2,M
*2+3; PAPER 2;"V": NEXT W
9780 GO TO 235
9800 FOR W=1 TO 60: PRINT AT X*2
+2+3; PAPER 2;"V"; AT X*2,Y*2+3;
"V": NEXT W: PRINT AT N*2,M*2+3;
PAPER 2;"V": FOR W=1 TO 60: P
INT AT 4*N-2*X,4*M-2*Y+3; "V": AT
4*N-2*X,4*M-2*Y+3; PAPER 2;"V":
NEXT W
9805 GO SUB 9400
9807 GO SUB 9300
9810 GO TO 5000
9990 FOR N=0 TO 7: READ S: POKE
USR "A"+N,S: NEXT N
9992 DATA 231,195,129,0,0,129,19
5,231
9995 RETURN

```

F. Delaquaize

#### REVERSI-DAMES (SPECTRUM 16 K)

Un très beau reversi avec plein de couleurs et des pions qui flashent avant de se placer. Les règles du jeu, s'il en est besoin sont expliquées dans la liste.



```

10 REM @CLERGEOT Bertrand 1983
12 GO SUB 9990
15 PAPER 4: INK 0: CLS : BORDE
4
200 GO SUB 5000
300 PAPER 4: INK 0: CLS : BORDE
4: CLS
95 PRINT
100 FOR N=1 TO 5
110 PRINT TAB 4; PAPER 7;" "
120 PRINT TAB 4; PAPER 7;" "
122 FOR N=1 TO 4
160 PRINT TAB 4; PAPER 7;" "
161 PRINT TAB 4; PAPER 7;" "
162 PRINT TAB 4; PAPER 7;" "
163 PRINT TAB 4; PAPER 7;" "
164 PRINT TAB 4; PAPER 7;" "
165 PRINT TAB 4; PAPER 7;" "
166 PRINT TAB 4; PAPER 7;" "
167 PAPER 0
168 DIM P(10,10)
172 FOR N=0 TO 1: FOR M=1 TO 5:
LET P(N*2+1,M*2-1)=1: LET P(10-N*
N*2-1,M*2-1)=5: PRINT AT N*4+2,M
*4+1; PAPER 5;"": PRINT AT 22-N
*4-2,M*4+3; PAPER 2;"": NEXT M
180 NEXT N
190 FOR N=0 TO 1: FOR M=1 TO 5:
LET P(N*2+2,M*2)=1: LET P(10-N*
2,M*2)=5: PRINT AT N*4+4,M*4+3;
PAPER 5;"": PRINT AT 22-N*4-4,M
*4+1; PAPER 2;"": NEXT M
200 NEXT N
210 INK 0: PAPER 4: FOR N=1 TO
10: PRINT AT N*2,2:N: PRINT AT 0
,N*2+3;CHR$(64+N): NEXT N
212 GO SUB 9400
220 INPUT "JE COMMENCE ? ";A$:
IF A$="O" THEN GO TO 5000
235 GO SUB 9300
238 BEEP .1,20
240 INPUT "A: enter B3 enter";A
$:B$: IF A$<>"FIN" AND B$<>"FIN"
THEN LET Y1=CODE A$(1)-64: LET
Y2=CODE B$(1)-64
242 IF A$="FIN" OR B$="FIN" THE
N GO TO 9000
250 LET X1=VAL A$(2 TO ): LET X
2=VAL B$(2 TO )
255 IF P(X1,Y1)<>1 THEN GO TO 2
40
260 IF P(X2,Y2)<>0 OR X2<1 OR X
2>10 OR Y2<1 OR Y2>10 OR X1<1 OR
X1>10 OR Y1<1 OR Y1>10 THEN GO
TO 240
270 IF X2-X1=-1 OR ABS (X2-X1)>
2 OR ABS (Y1-Y2)>2 OR ABS (X1-X2
)>2 OR ABS (Y1-Y2)>2 THEN GO TO 240
280 IF ABS (Y1-Y2)=2 AND P((X2-
X1)/2+X1,(Y2-Y1)/2+Y1)<>8 THEN G
O TO 240
290 PRINT AT X1*2,Y1*2+3;" ";AT
X2*2,Y2*2+3; PAPER 5;" "
300 LET P(X1,Y1)=0: LET P(X2,Y2
)=1: IF ABS (Y1-Y2)=2 THEN LET P
((X2-X1)/2+X1,(Y2-Y1)/2+Y1)=1: P
RINT AT (X2-X1)/2+X1,(Y2-Y1)/2+Y
1+3; PAPER 5;"": GO TO 240
305 FOR N=9 TO 2 STEP -1
310 FOR M=9 TO 2 STEP -1
320 IF P(N,M)=1 AND P(N-1,M-1)=
0 AND P(N+1,M+1)=0 THEN LET P(N+
1,M+1)=8: LET P(N,M)=8: LET P(N-
1,M-1)=0: LET X=N-1: LET Y=M-1:
GO TO 9000
330 IF P(N,M)=1 AND P(N+1,M+1)=
0 AND P(N-1,M-1)=0 THEN LET P(N+
1,M+1)=8: LET P(N,M)=8: LET P(N-
1,M-1)=0: LET X=N+1: LET Y=M+1:
GO TO 9000
340 IF P(N,M)=1 AND P(N-1,M+1)=
0 AND P(N+1,M-1)=0 THEN LET P(N+
1,M-1)=8: LET P(N,M)=8: LET P(N-
1,M+1)=0: LET X=N-1: LET Y=M+1:
GO TO 9000
350 IF P(N,M)=1 AND P(N+1,M-1)=
0 AND P(N-1,M+1)=0 THEN LET P(N+
1,M+1)=8: LET P(N,M)=8: LET P(N+
1,M-1)=0: LET X=N+1: LET Y=M-1:
GO TO 9000
360 NEXT M: NEXT N
370 LET K=0
380 FOR N=1 TO 9
390 FOR M=10 TO 1 STEP -1
400 IF M=1 THEN GO TO 9500
410 IF M=10 THEN GO TO 9650
420 IF P(N,M)=0 AND P(N+1,M-1)=
0 THEN GO TO 9750
430 IF P(N,M)=0 AND P(N+1,M+1)=
0 THEN GO TO 9700
440 NEXT M
450 NEXT N
460 IF K=1 THEN GO TO 9000
470 LET K=1

```

```

5160 GO TO 5080
5200 DRAW 0,5: DRAW 70,42: PLOT
0,0: DRAW 255,0: DRAW 0,5: DRAW
255,0: PLOT 255,5: DRAW -70,42
5310 FOR N=1 TO 10: PLOT N*25,5,
5410 DRAW 11.5*N+70-N*25.5,42: NEX
T N
5520 PLOT 20,17: DRAW 215,0: PLO
T 38,28: DRAW 177,0: PLOT 52,37:
DRAW 151,0: PLOT 62,43: DRAW 13
0,0: PLOT 69,47: DRAW 117,0
5630 PRINT AT 6,6; INK 1;"
5730 PRINT AT 13,6; INK 1;"
5830 FOR N=7 TO 12: PRINT AT N,6
: INK 1;"
5940 NEXT N
6040 INK 2: PLOT 42,133: DRAW 17
0,0: DRAW 0,-75: DRAW -172,0: DR
AW 0,75
6140 PLOT 45,130: DRAW 166,0: DR
AW 0,-69: DRAW -166,0: DRAW 0,69
6250 PRINT AT 8,8; INK 0;"CLERGE
OT B. 1983"
6350 PRINT AT 10,10; INK 0;"Reve
lame"
6470 IF INKEY$="" THEN GO TO 607
6572 BORDER 0: INK 1: PAPER 0: C
LS : OVER 1
6640 OVER 0: PLOT 40,50: DRAW 17
5,0: DRAW 0,95: DRAW -175,0: DR
AW 0,-95
6740 OVER 0: PLOT 37,47: DRAW 18
1,0: DRAW 0,101: DRAW -181,0: DR
AW 0,-101
6850 INK 5
6950 PRINT AT 6,8;"REGLE DU JEU"
7071 PRINT AT 8,6;"le déplacemen
t: AT 9,6;"est celui des dames";
AT 10,6;"mais une piece prise";
AT 11,6;"devient alliee";AT 12,6
7190 IF INKEY$<>" " THEN GO TO 70
80
7290 IF INKEY$="" THEN GO TO 708
7310 PRINT AT 6,6;"Vous devez ca
pturer";AT 7,6;"les pieces rouge
s";AT 8,6;"le jeu fini lorsque";
AT 9,6;"la situation est";
AT 10,6;"bloquee: gagne celui";
AT 11,6;"qui est majoritaire";
7410 IF INKEY$<>" " THEN GO TO 71
10
7520 IF INKEY$="" THEN GO TO 712
7630 PRINT AT 6,6;"lorsque vous
avez";AT 7,6;"pris, vous rejou
ez";AT 8,6;"vous n'etes pas";
AT 9,6;"obligé de prendre";AT
10,6;"
7760 IF INKEY$<>" " THEN GO TO 72
80
7820 IF INKEY$="" THEN GO TO 721
7930 RETURN
8000 GO SUB 9300: IF C<20 THEN P
RINT AT 8,9; FLASH 1; PAPER 6;"J
AI GAGNE"
8120 IF C>20 THEN PRINT AT 0,5;
FLASH 1; PAPER 6;" VOUS AVEZ GA
GNE"
8230 IF C=20 THEN PRINT AT 0,4;
FLASH 1; PAPER 6;" LA PARTIE EST
NULLE"
8340 STOP
8400 LET C=0: FOR N=1 TO 10: FOR
M=1 TO 10: IF P(N,M)=1 THEN LET
C=C+1
8510 NEXT M: NEXT N
8620 PRINT AT 1,27; BRIGHT 1;C:
PAPER 4: INK 5;" ";AT 3,27;BRI
GHT 1; INK 0;40-C; INK 2;" "
8730 RETURN
8840 FOR N=7 TO 0 STEP -1: INK N
: PLOT 32,167: DRAW 167,0: DRAW
0,-167: DRAW -167,0: DRAW 0,167:
NEXT N
8940 RETURN
9050 IF P(N-1,M-1)=1 OR (P(N+1,M
-1)=1 AND P(N-1,M+1)=0) OR (P(N-
1,M+1)=1 AND P(N+1,M-1)=0) THEN
LET D=1
9150 RETURN
9250 IF P(N-1,M+1)=1 OR (P(N+1,M
+1)=1 AND P(N-1,M-1)=0) OR (P(N-
1,M-1)=1 AND P(N+1,M+1)=0) THEN
LET D=1
9350 RETURN
9400 IF P(N+1,M+1)<>8 OR P(N,M)<
>0 THEN GO TO 5130
9510 GO TO 9710
9650 IF P(N+1,M-1)<>8 OR P(N,M)<
>0 THEN GO TO 5130
9760 GO TO 9760
9800 LET D=0: GO SUB 9500: IF D=
1 AND K=0 THEN GO TO 5130
9910 LET P(N,M)=8: LET P(N+1,M+1
)=0: FOR W=0 TO 60: PRINT AT N*2

```

B. Clergeot

Un jeu de réflexion original et intéressant. L'ordinateur génère une grille de 10x10 remplie de chiffres de 1 à 9. Les règles du jeu sont expliquées dans le programme, il est essentiel que chaque joueur joue dans la colonne ou la ligne dans laquelle l'autre vient de jouer. L'ordinateur joue toujours le chiffre le plus grand.

```

100 REM
200 REM
300 REM
400 REM
500 REM
600 REM
700 REM
800 REM
900 REM
1000 REM
1100 REM
1200 REM
1300 REM
1400 REM
1500 REM
1600 REM
1700 REM
1800 REM
1900 REM
2000 REM
2100 REM
2200 REM
2300 REM
2400 REM
2500 REM
2600 REM
2700 REM
2800 REM
2900 REM
3000 REM
3100 REM
3200 REM
3300 REM
3400 REM
3500 REM
3600 REM
3700 REM
3800 REM
3900 REM
4000 REM
4100 REM
4200 REM
4300 REM
4400 REM
4500 REM
4600 REM
4700 REM
4800 REM
4900 REM
5000 REM
5100 REM
5200 REM
5300 REM
5400 REM
5500 REM
5600 REM
5700 REM
5800 REM
5900 REM
6000 REM
6100 REM
6200 REM
6300 REM
6400 REM
6500 REM
6600 REM
6700 REM
6800 REM
6900 REM
7000 REM
7100 REM
7200 REM
7300 REM
7400 REM
7500 REM
7600 REM
7700 REM
7800 REM
7900 REM
8000 REM
8100 REM
8200 REM
8300 REM
8400 REM
8500 REM
8600 REM
8700 REM
8800 REM
8900 REM
9000 REM
9100 REM
9200 REM
9300 REM
9400 REM
9500 REM
9600 REM
9700 REM
9800 REM
9900 REM
10000 REM

```

QUI GAGNE

D. GUMY

(C) JUIN 1983

MACRO-LETTERS

REGLES DU JEU

INITIALISATION

1000 GOSUB 1000

1100 REM

1200 GOSUB 1500

1300 REM

1400 REM

1500 REM

1600 REM

1700 REM

1800 REM

1900 REM

2000 REM

2100 REM

2200 REM

2300 REM

2400 REM

2500 REM

2600 REM

2700 REM

2800 REM

2900 REM

3000 REM

3100 REM

3200 REM

3300 REM

3400 REM

3500 REM

3600 REM

3700 REM

3800 REM

3900 REM

4000 REM

4100 REM

4200 REM

4300 REM

4400 REM

4500 REM

4600 REM

4700 REM

4800 REM

4900 REM

5000 REM

5100 REM

5200 REM

5300 REM

5400 REM

5500 REM

5600 REM

5700 REM

5800 REM

5900 REM

6000 REM

6100 REM

6200 REM

6300 REM

6400 REM

6500 REM

6600 REM

6700 REM

6800 REM

6900 REM

7000 REM

7100 REM

7200 REM

7300 REM

7400 REM

7500 REM

7600 REM

7700 REM

7800 REM

7900 REM

8000 REM

8100 REM

8200 REM

8300 REM

8400 REM

8500 REM

8600 REM

8700 REM

8800 REM

8900 REM

9000 REM

9100 REM

9200 REM

9300 REM

9400 REM

9500 REM

9600 REM

9700 REM

9800 REM

9900 REM

10000 REM

```

1000 LET XX=28
1100 LET YY=19
1110 LET A$="?"
1120 GOSUB 1300
1125 SLOW
1130 LET A$="CONCEPTION : DIDIER GUMY"
1140 LET Y=LEN A$
1150 FOR I=Y TO 1 STEP -1
1160 FOR J=1 TO 31-Y+1
1170 PRINT AT 19,J;A$(I)
1180 PRINT AT 19,J-1;" "
1190 NEXT J
1200 NEXT I
1210 PRINT AT 21,0;"POUR CONTINU
ER PRESSEZ SUR
1215 PAUSE 4E4
1220 IF INKEY$="" THEN GOTO 1220
1230 IF INKEY$(">")"C" THEN GOTO 12
20
1250 CLS
1260 RETURN
1300 REM
1305 REM
1310 LET Z$="RAND4("
1315 LET XC=CODE A$(A)
1320 GOSUB 1340
1325 LET XX=XX+8
1330 NEXT A
1335 RETURN
1340 FOR Y=YY TO YY-7 STEP -1
1345 LET X$=CHR$(PEEK(7680+8*X
O+YY-Y))
1350 FOR X=1 TO 8
1355 IF X$(Z$(X)) THEN GOTO 1370
1360 PLOT XX+X-1,Y
1365 LET X$=CHR$(CODE X$-CODE Z
$(X))
1370 NEXT X
1375 NEXT Y
1380 RETURN
1500 REM
1510 PRINT AT 0,10;"QUI GAGNE"
1520 PRINT AT 1,9;"-----"
1530 PRINT "VOUS DEVEZ CHOISIR U
N CHIFFRE, DANS LA GRILLE; SI JE
VOUS AVANT, VOUS DEVEZ JO
UER SUR LA MEME LIGNE OU SUR L
A MEME COLONNE QUE MOI."
1535 PRINT "LE SCORE LE PLUS HAU
T GAGNE, EVIDEMENT, BONNE CH
ANCE."
1540 PRINT AT 21,0;"POUR CONTINU
ER PRESSEZ SUR
1545 PAUSE 4E4
1550 IF INKEY$="" THEN GOTO 1550
1560 IF INKEY$(">")"C" THEN GOTO 15
50
1570 CLS
1580 RETURN
2000 REM
2010 LET CH=INT(RND*10)+1
2020 LET CV=INT(RND*10)+1
2030 RETURN
2100 REM
2110 FOR N=9 TO 0 STEP -1
2120 FOR X=1 TO 10
2130 IF A(CH,X)=N THEN GOTO 2170
2140 IF A(X,CV)=N THEN GOTO 2200
2150 NEXT X
2160 NEXT N
2170 LET CC=CH
2180 LET DD=X
2190 RETURN
2200 LET CC=X
2210 LET DD=CV
2220 RETURN
2300 REM
2310 PRINT AT 18,0;" LA PARTI
E EST TERMINEE"
2320 IF T1>T2 THEN PRINT AT 18,0
"VOUS AVEZ GAGNE"
2330 IF T1<T2 THEN PRINT AT 18,0
"VOUS AVEZ PERDU"
2340 IF T1=T2 THEN PRINT AT 18,0
"PARTIE NULLE"
2350 PRINT AT 20,0;" UNE NOUV
ELLE PARTIE (O/N)";"?"
2360 IF INKEY$="" THEN GOTO 2560
2370 IF INKEY$="O" THEN GOTO 260
0
2380 IF INKEY$="N" THEN GOTO 265
0
2500 GOTO 2560
2510 CLEAR
2520 CLS
2530 GOTO 130
2540 CLS
2550 GOTO 9990
2560 REM
2570 SAVE "QUI GAGNE"
2580 GOSUB 1000
2590 PRINT AT 18,0;"VOULEZ-VOUS
LES REGLES DU JEU";TAB 8;"(O/N)
";"?"
2610 IF INKEY$="" THEN GOTO 2740
2620 IF INKEY$="O" THEN GOTO 278
0
2630 IF INKEY$="N" THEN GOTO 280
0
2670 GOTO 2740
2680 CLS
2690 GOSUB 1500
2700 CLS
2710 GOTO 130
2720 REM
2730 PRINT AT 20,10;"MERCI"
2740 STOP

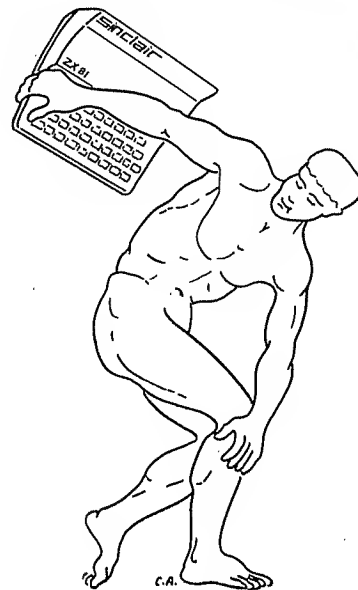
```

Une belle réalisation en langage machine - liste garantie réverifiée 3 fois ! - Ce jeu utilise la manette de jeu simplifiée dont la construction a été proposée par ORDI-5 ; mais le jeu fonctionne aussi si l'on utilise les touches du clavier (toutes les touches sont efficaces, même BREAK). Attention la routine est rapide.

```

50 REM
100 POKE 16418,0
200 FOR A=0 TO 41
300 PRINT "BOWLING";
400 PRINT "BOWLING";
500 NEXT A
600 PRINT "APPUYEZ SUR U
VE TOUCHE"
70 IF INKEY#="" THEN GOTO 70
80 LET A#=""
NG BOWLI
900 CLS
920 POKE 16418,2
950 LET L=USR 16514
1000 PRINT A#
110 PRINT AT 10,3;"NOMBRE DE JO
EURS ? (1 A 9)"
1200 LET I#="INKEY#"
1300 IF I#="" THEN GOTO 020
1350 IF CODE I#<29 OR CODE I#>37
THEN GOTO 120
1360 CLS
1370 PRINT A#
1400 LET N=VAL I#
1450 DIM S(N)
1500 DIM B$(N,20)
1600 FOR A=1 TO N
170 PRINT AT 6+A,0;"NOM DU ";A;
"JOUEUR:";
180 INPUT B$(A)
1900 PRINT B$(A)
2000 NEXT A
2100 CLS
2200 PRINT A#
2300 PRINT "NOMBRE DE PARTIE
S PAR JOUEUR:";
2400 INPUT P
2500 CLS
2600 PRINT A#
2700 PRINT "CHAQUE JOUEUR DI
POSE DE S
BOULES PAR PARTI
"
2800 PRINT "APPUYEZ SUR UNE
TOUCHE"
2950 IF INKEY#="" THEN GOTO 295
3000 FOR A=1 TO N
3100 CLS
3200 PRINT A#
3300 PRINT "A VOUS,";B$(A)
3400 PRINT
3500 PRINT "NIVEAU DE DIFFICULTE
?(1 A 255)"
3600 INPUT D
3700 IF D<1 OR D>255 THEN GOTO 3
3800 POKE 16417,255-D
3850 FOR B=1 TO P
3900 PRINT "APPUYEZ SUR UNE
TOUCHE POUR COMMENCER LE JEU
"
3950 IF INKEY#="" THEN GOTO 395
4000 POKE 16507,0
4100 CLS
4200 LET L=USR 16765
4300 LET S(A)=S(A)+PEEK 16507
4400 CLS
4450 IF B=P THEN GOTO 480
4500 PRINT AT 5,2;B;"PARTIE:";P
PEEK 16507;"QUILLE";
4550 IF PEEK 16507>1 THEN PRINT
"S";
4600 PRINT AT 2;"PRET POUR LA "
;B+1;"PARTIE ?";
4700 NEXT B
4800 PRINT AT 5,2;"BRAVO ";B$(A)
;TAB 2;"VOUS AVEZ TOTALISE ";S(A)
;"QUILLE";
4810 IF S(A)>1 THEN PRINT "S"
4900 FOR Z=0 TO 100
5000 NEXT Z
5100 CLS
5200 NEXT A
5250 PRINT A#
5300 PRINT
5350 FOR A=1 TO N
5450 PRINT B$(A);S(A);"QUILLE";
5470 IF S(A)>1 THEN PRINT "S"
5500 PRINT
5600 NEXT A
5700 PRINT "VOULEZ-VOUS REJO
UER ? (O/N)"
5800 IF INKEY#="" THEN GOTO 580
5900 IF INKEY#="O" THEN GOTO 630
6000 PRINT "AU REVOIR..."
6100 STOP
6200 SAVE "BOWLING"
6300 CLS
6400 RUN

```



```

16514,42,12,64,17,221,0,25,205
16522,211,64,17,137,0,25,205
16529,211,64,17,104,25,25,14
16536,15,205,4,355,17,205,25,14
16544,12,2,170,4,355,17,205,25,14
16551,355,13,355,2,25,109,233,64
16558,355,13,355,2,25,109,233,64
16565,355,13,355,2,25,109,233,64
16572,355,13,355,2,25,109,233,64
16579,355,13,355,2,25,109,233,64
16586,355,13,355,2,25,109,233,64
16593,355,13,355,2,25,109,233,64
16600,355,13,355,2,25,109,233,64
16607,355,13,355,2,25,109,233,64
16614,355,13,355,2,25,109,233,64
16621,355,13,355,2,25,109,233,64
16628,355,13,355,2,25,109,233,64
16635,355,13,355,2,25,109,233,64
16642,355,13,355,2,25,109,233,64
16649,355,13,355,2,25,109,233,64
16656,355,13,355,2,25,109,233,64
16663,355,13,355,2,25,109,233,64
16670,355,13,355,2,25,109,233,64
16677,355,13,355,2,25,109,233,64
16684,355,13,355,2,25,109,233,64
16691,355,13,355,2,25,109,233,64
16698,355,13,355,2,25,109,233,64
16705,355,13,355,2,25,109,233,64
16712,355,13,355,2,25,109,233,64
16719,355,13,355,2,25,109,233,64
16726,355,13,355,2,25,109,233,64
16733,355,13,355,2,25,109,233,64
16740,355,13,355,2,25,109,233,64
16747,355,13,355,2,25,109,233,64
16754,355,13,355,2,25,109,233,64
16761,355,13,355,2,25,109,233,64
16768,355,13,355,2,25,109,233,64
16775,355,13,355,2,25,109,233,64
16782,355,13,355,2,25,109,233,64
16789,355,13,355,2,25,109,233,64
16796,355,13,355,2,25,109,233,64
16803,355,13,355,2,25,109,233,64
16810,355,13,355,2,25,109,233,64
16817,355,13,355,2,25,109,233,64
16824,355,13,355,2,25,109,233,64
16831,355,13,355,2,25,109,233,64
16838,355,13,355,2,25,109,233,64
16845,355,13,355,2,25,109,233,64
16852,355,13,355,2,25,109,233,64
16859,355,13,355,2,25,109,233,64
16866,355,13,355,2,25,109,233,64
16873,355,13,355,2,25,109,233,64
16880,355,13,355,2,25,109,233,64
16887,355,13,355,2,25,109,233,64
16894,355,13,355,2,25,109,233,64
16901,355,13,355,2,25,109,233,64
16908,355,13,355,2,25,109,233,64
16915,355,13,355,2,25,109,233,64
16922,355,13,355,2,25,109,233,64
16929,355,13,355,2,25,109,233,64
16936,355,13,355,2,25,109,233,64
16943,355,13,355,2,25,109,233,64
16950,355,13,355,2,25,109,233,64
16957,355,13,355,2,25,109,233,64
16964,355,13,355,2,25,109,233,64
16971,355,13,355,2,25,109,233,64
16978,355,13,355,2,25,109,233,64
16985,355,13,355,2,25,109,233,64
16992,355,13,355,2,25,109,233,64
17000,355,13,355,2,25,109,233,64
17007,355,13,355,2,25,109,233,64
17014,355,13,355,2,25,109,233,64
17021,355,13,355,2,25,109,233,64
17028,355,13,355,2,25,109,233,64
17035,355,13,355,2,25,109,233,64
17042,355,13,355,2,25,109,233,64
17049,355,13,355,2,25,109,233,64
17056,355,13,355,2,25,109,233,64
17063,355,13,355,2,25,109,233,64
17070,355,13,355,2,25,109,233,64
17077,355,13,355,2,25,109,233,64
17084,355,13,355,2,25,109,233,64
17091,355,13,355,2,25,109,233,64
17098,355,13,355,2,25,109,233,64
17105,355,13,355,2,25,109,233,64

```



# LE LASER 200 1280<sup>F</sup> TTC



## L'INCROYABLE MICRO-ORDINATEUR COULEUR SECAM !

- Microprocesseur Z 80 A
- Langage Microsoft Basic
- Affichage direct antenne télé SECAM
- Clavier 45 touches pleine écriture,  
+ clef d'entrée, + graphismes,  
+ bip sonore anti-erreurs...

- Texte + graphismes mixables 9 couleurs
- Edition et correction plein écran
- Son incorporé
- Toutes options : extension + 16 K + 64 K,  
interface imprimante, imprimante, stylo optique,  
manettes, jeux, modem, disquettes...



## VIDEO TECHNOLOGIE FRANCE

19, rue Luisant 91310 Monthléry  
Tél. (6) 901.93.40 - Télex : SIGMA 180114

### BON DE COMMANDE

A retourner à : VIDEO TECHNOLOGIE - 19, rue Luisant - 91310 Monthléry - Tél. (6) 901.93.40 - Télex SIGMA 180114

Je désire recevoir :

☐ **Version A**

Micro-ordinateur couleur SECAM LASER 200 ..... 990 F TTC

Kit d'accessoires :

- Modulateur SECAM incorporé
- + Transfo 220 V 50 HZ
- + 3 interfaces : câble télé, câble vidéo, câble lecteur K7
- + Livre utilisateur Basic en français, 150 pages
- + Livrets techniques en français
- + Cassette
- + Garantie 1 an, pièces et main-d'œuvre

Le kit complet ..... 290 F TTC  
1.280 F TTC

#### Extensions - Périphériques - Interfaces

- ☐ Extension de mémoire 16 K RAM (soit 20 K disponibles) 540 F TTC
- ☐ Extension de mémoire 64 K RAM (soit 68 K disponibles)  
(livraison fin octobre) ..... 990 F TTC
- ☐ Lecteur de cassette DR 10 ..... 490 F TTC
- ☐ Interface d'imprimante « Centronics » ..... 290 F TTC
- ☐ Imprimante 4 couleurs (livraison fin septembre) ... 2.360 F TTC
- ☐ Manettes de jeux (la paire) (livraison fin septembre) . 290 F TTC
- ☐ Stylo lumineux (livraison fin octobre) ..... N.C.
- ☐ Interface disquette (livraison fin octobre) ..... N.C.

TOTAL DE MA COMMANDE : ..... F TTC

Nom .....

Prénom .....

N° ..... Rue .....

Ville .....

Code Postal [ ] [ ] [ ] [ ] [ ] [ ]

Je choisis de payer le total de ma commande :

- ☐ Au comptant, par CCP, chèque bancaire ou mandat, à l'ordre de VIDEO TECHNOLOGIE FRANCE.
- ☐ Contre-remboursement au transporteur, moyennant une taxe de 60 F.

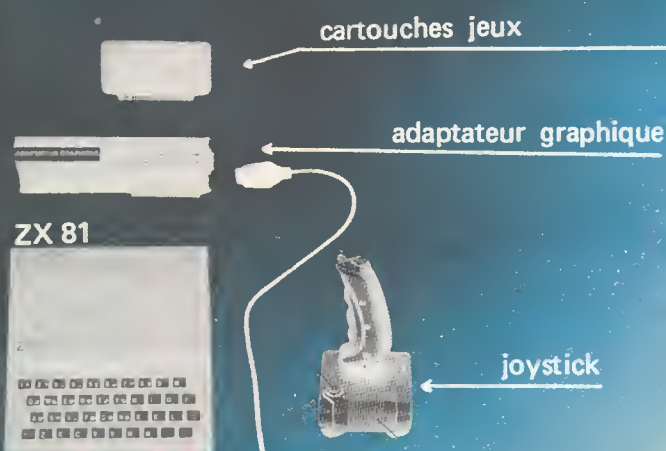
Signature

Au cas où je ne serais pas entièrement satisfait, je suis libre pendant un délai de 15 jours, de retourner à mes frais dans son emballage d'origine le matériel que j'aurai reçu et je serai intégralement remboursé des sommes que j'aurai versées.

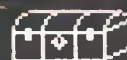


# VTR Software

54, rue Ramey 75018 PARIS téléphone : 252.87.97



INTERCEPTEUR



MINEUR



DESTRUCTEUR



DINKEUR



CROQUEUR



ENVAHISSEUR



TAMPONNEUR

ET BIEN  
D'AUTRES  
A VENIR

## CARTOUCHES DISPONIBLES ACTUELLEMENT

DEMANDEZ NOTRE CATALOGUE  
GÉNÉRAL SOFTWARE  
CARTOUCHES ET CASSETTES  
pour  
ZX 81, ORIC, SPECTRUM, VIC 20  
COMMODORE 64

nos cartouches de jeux travaillent en haute résolution  
grâce à l'adaptateur graphique sur un ZX 81 de base 1K.

elles sont également compatibles avec la carte SAM (cou-  
leur) et la carte VTR (son).

le jeu est alors en haute résolution couleur et sonore !

VOUS CONNECTEZ LA CARTOUCHE  
et VOUS JOUEZ !

Une sélection des meilleurs logiciels...  
Les noms les plus prestigieux

- CASES COMPUTER SIMULATIONS
- Mr. CHIP SOFTWARE
- HEWSON CONSULTANTS
- IMAGINE
- INTERCEPTOR MICRO'S
- J K GREY ENTERPRISES
- LLAMASOFT
- MARTECH GAMES
- MIKROGEN
- QUICK SILVA
- R and R SOFTWARE
- I J K SOFTWARE
- NEW GENERATION SOFTWARE

Les produits de VTR Software sont disponibles  
dans les points de vente VTR Informatique.

Magasin de vente: Même adresse

Horaires : 10 h - 30 - 13 h - 30 et 15 h - 19 h.

Jours d'ouverture : du mardi au samedi inclus.

Métro : Jules Joffrin ou Marcadet Poissonniers

VTR Software

est un Département de Vidéo Télémat Report Sarl

Pour recevoir notre catalogue, remplissez le coupon ci-dessous et retournez-le accompagné  
de 5 F en timbres, à VTR Software, 54 rue Ramey, 75018 PARIS.

DEMANDE DE CATALOGUE

Nom : \_\_\_\_\_ Prénom : \_\_\_\_\_

Adresse : \_\_\_\_\_

Code Postal : \_\_\_\_\_